In this special issue, we address some recent, state-of-the-art, approachs for querying, approximating, estimating and maintaining large data sets. In many of these cases, users would like to start with summary or approximate answers that would guide them in the direction of their quest. This approximate answer may be useful in itself, or may be a first step to more detailed and more specific drill down queries. Large data sets are often characterized by multiple attributes that refer to various characteristics of the data. As a result, large collections of data are often summarized in a data cube, which has frequently been used for OLAP applications. Many of the papers in this special issue specifically address the issue of approximating and maintaining information in large data cubes.

Poosala, Ganti and Ioannidis have been involved in an effort to build an efficient data analysis system called Aqua. In the first paper they describe how to use histograms to approximate queries when precise answers are not necessary and early feedback is helpful. Given the exploratory nature of many OLAP applications, precise answers often result in answers of no particular interest to the users. However, for approximate answers, users need to be provided with some measure of confidence in the answers. Hence, in this paper, histograms are enhanced to provide quality guarantees on the approximate answers. Furthermore, the authors provide an efficient technique for selecting histograms that result in the most accurate answers while using small amounts of space.

In the second paper, Barbara and Wu describe how Quasi-Cubes can be used to compress data cubes and provide approximate answers with guaranteed bounds on the errors. They also discuss a variety of applications that would benefit from such approximate answers inclulding data marts (light-weight data warehouses), and approximate data mining. In such data mining applications, exploratory data analysis tries to determine which factors have the most influence on the data values. Alternatively, decision trees can be used as a classification technique to forecast characteristics of various data items. In such applications fast approximate answers have the potential of resulting in significant benefits in efficiency.

Johnson and Shasha address the problem of maintaining large data cubes. In particular, they present a new storage structure called cube forests. They present algorithms for the querying and updating of cube forests and also describe how to perform batch updates. An implementation of cubes forests is described and a simulation study using the TPC-D benchmark shows the significant reduction of costs for batch updates.

Geffner, Riedewald, Agrawal and El Abbadi describe two alternative approaches for storing data cubes. These approaches present a tradeoff between the cost of querying versus the cost of updating. In the relative prefix sum approach query cost is kept constant but updates are proportional to the square root of the size of the cube. This approach is useful for applications where queries are the dominant application, but the data still needs to be updated. In the second approach, the dynamic data cubes, both queries and updates are have logarithmic cost, which makes this data structures especially appropriate for speculative, what-if analytical processing applications.

Chaudhari and Motwani propose using sampling as a method for estimating query results. In fact, they argue that for many data mining and OLAP applications, sampling must be supported on the result of an arbitrary query, not just on the data itself. For this purpose, they explore the major impediments to implementing sampling as a primitive relational operation. They demonstrate that auxiliary information such as histograms can be leveraged to facilitate efficient sampling.

The papers in this issue represent on-going research and implementation efforts. They pose several open and interesting problems and hence I hope this special issue will lead to more research and investigations in this increasingly significant area. Current applications range from traditional databases to OLAP and data mining and potentially are useful in diverse domains such as digital libraries, biological computing, as well as enviromental and economic applications.