Chapter 1

Introduction

Advances in wireless technology enable connectivity in mobile space, making a wide-scale adoption of mobile information systems feasible. Moreover, mobile devices are constantly getting smaller, cheaper and more powerful. Laptops can now easily store and process large databases, something once possible only by heavyweight stationary servers. The power and capacity of PDAs are continuously increasing, being already capable of performing a number of database tasks quite efficiently.

Organizations deploy mobile applications because substantial business benefits can be safely assumed. Nowadays, ubiquitous data access in an extremely competitive environment is success critical. Data is being generated with a tremendous rate and decision making based on the generated data has to be immediate.

Consequently, the personal and terminal mobility have become indispensable components of the highly efficient workforce. It has become a common approach to transform any location and situation into a job office. Thus a manager, while traveling in a car or on a plane or sitting on a beach, can create a job office on the spot and can access necessary information to complete its tasks. However in such conditions, network connectivity cannot always be guaranteed, necessitating offline solutions as well.

Data warehousing is a traditional decision support tool. Based on multidimensional models, it manages huge amounts of aggregated, hierarchical data. Increased data volumes and accelerating update speeds are fundamentally changing the role of the modern data warehouse (DW) too. Data warehousing is increasingly used not only for strategic, but for operative decision making as well. More data, coming in faster and requiring immediate conversion into decisions means that organizations are confronting the need for right-time (active) data warehousing.

Evidently, enabling mobile, portable devices to participate in such decision support systems is a fundamental requirement. The research field of mobile online analytical processing (mOLAP) brings the two aforementioned research and application areas together. The term mOLAP encompasses all necessary technologies for mobile information systems, which enable multidimensional data access to users that carry a portable, mobile device. It shall be shown in the rest of the thesis that mOLAP lies within the borders of several different research areas, demanding a rather unconventional interplay between them.

1.1 Problem Statement

Data broadcast can be managed with three different modes: *on-demand, push-based* and *hybrid*. In on-demand broadcast, clients issue explicit requests for data. If multiple clients request the same data at approximately the same time, the server may match these requests and broadcast data only once. In push-based systems, the server employs point-to-multipoint communication and sends data items from the server to clients in the absence of explicit client requests. In order to achieve that, the server maintains a broadcast schedule, which determines the order and the frequency in which data items are broadcast. Hybrid dissemination systems, combine broadcast push and pull, by assigning popular data items to a push channel and unpopular to a pull channel.

Efficient and robust mOLAP cannot be provided by general data broadcast, regardless of the mode for the following reasons:

1. *Data semantics*: Content and characteristics of broadcast data items are not considered. On the contrary though, multidimensional data cubes are semantically connected to each other.

2. *Thin/Fat clients*: Due to the underlying assumption of thin clients, transmitted data is processed for direct end usage. As mobile devices become increasingly powerful, a lot of space for load balancing is open. Client local processing, not only improves the performance of the entire system, but enables offline functionality as well.

3. *OLAP end user behavior*: OLAP end users typically navigate through the requested data cubes, performing typical OLAP operations such as roll-ups or drill-downs. This has to be considered in the design of the dissemination system in order to avoid additional transmissions.

4. *Offline functionality*: The offline aspect is completely absent, again due to the assumption of a thin client population. However, the entire concept of mOLAP systems is founded on providing offline functionality. This is crucial when considering that mobile devices are not permanently connected to a network.

Beyond the above observations, different broadcast modes exhibit additional shortcomings. On-demand systems are notoriously not scalable, since they fail to keep up with growing client population and consequently growing number of incoming requests. Although they can efficiently serve a limited number of end users, they are definitely unsuitable as this number increases.

Scalability is addressed by push-based data broadcast. Nevertheless, conventional push-based broadcast is also unsuitable for mOLAP for the following reasons:

1. *Data population*: The number of handled data items in mOLAP is not limited, which is a common assumption of existing push-based systems. Data items in mOLAP are query answers. As the number of possible queries is practically infinite, a typical broadcast schedule cannot be employed.

2. *Data item size*: Data items do not occupy relatively small size (e.g., web pages), which is also a common assumption of existing push-based systems. Multidimensional data cubes are items order of magnitude bigger than web pages.

Moreover, not only in mOLAP but generally, push-based systems unnecessarily consume bandwidth when the number of incoming requests is relatively low.

Hybrid dissemination systems suffer more or less from the aforementioned observations. Although they are theoretically self adaptive, by dynamically assigning hot data items to push channels and cold data items to pull channels, in practice this incurs increased complexity and maintenance overhead.

Evidently, efficient and robust mOLAP cannot be provided by existing generic broadcast architectures. We argue that mOLAP requires an entirely different architecture, which considers and exploits the characteristics of multidimensional data and the behavior of OLAP end users.

A mOLAP architecture should exhibit, apart from efficiency in performance, scalability, self adaptiveness to workload fluctuations, client-server load balancing, online and offline functionality, independence of query distribution and data cube physical implementation.

1.2 Contribution

Considering the issues highlighted in the problem statement, the main contribution of this thesis is the presentation and performance evaluation of the Force Clustering OLAP Scheduler (FCLOS), a mobile information system explicitly designed for mOLAP. *FCLOS* exhibits the following features:

- *Offline functionality*: *FCLOS* clients typically receive sub-cubes which contain data at the most detailed view of the participating dimensions. This happens because the server maps incoming queries to the Data Cube Lattice, which eliminates less detailed views according to hierarchical levels. In the case of disconnections, end users retain advanced offline functionality.

- *Scalability*: *FCLOS* performance does not deteriorate, if the number of end clients increases. On the contrary, the subsumption exploitation results in enhancement of its performance, as the client population grows.

- *Self adaptiveness*: *FCLOS* reacts to explicit client queries and does not employ any fixed broadcast program. In this sense, it is classified as an on-demand architecture. However, the subsumption based handling of incoming queries and the extensive usage of broadcast resembles a push-based architecture. Practically, on the one hand, if the incoming load is relatively low, *FCLOS* exhibits the advantage of on-demand systems, namely that no broadcast program makes any unnecessary usage of the available bandwidth. On the other hand, if the load increases, due to the subsumption exploitation, *FCLOS* exhibits the main feature of push-based systems, scalability.

- *Load balancing*: *FCLOS* provides different ways of client-server load balancing. The fundamental idea in *FCLOS* is to send to a client a sub-cube that is either exactly the requested one or alternatively an ancestor sub-cube, namely a sub-cube that can be locally processed in order to produce the requested one. Since there might be more than one ancestor, the choice of the transmitted ancestor directly influences the load imposed on the client. Moreover, beyond the choice of the conceptual sub-cube, the choice of the physical structure does also provoke an indirect load balancing. The reason for that is that query execution time on local data is highly dependent on the compression and the indexing scheme of the physical structure. More compressed physical structures generate less traffic but incur increased local processing. Finally, it is possible to load balance with different query mappings to aggregation lattices.

- *Efficiency*: According to extensive experimental evaluation, *FCLOS* outperforms previous approaches on all criteria: query access time, energy consumption overhead and generated traffic. The degree of optimization is highly dependent on the used dataset, the query workload and distribution, and network environment. However, for our default scenario using a real data mart, *FCLOS* exhibits an optimization of more than 65% on the relevant counts.

- *Query distribution independence*: In order to reduce the number of handled items, incoming queries are mapped to the corresponding sub-cubes of an aggregation lattice. This results in a coarse-grained querying mode, and eventually *FCLOS* remains unaffected by varying query distributions.

- *Physical structure independence*: Scheduling decisions are completely separated from the size of the objects being handled. *FCLOS* considers dimensionality instead of size, which is independent of the physical implementation. This not only allows the selection between different data cube physical structures, but constitutes the architecture extendable as well. Any future data cube physical structure can be seamlessly incorporated.

The secondary contribution of the thesis is the Query and Disseminate under Global View (QDGV) protocol, which extends scenarios, in which clients exclu-

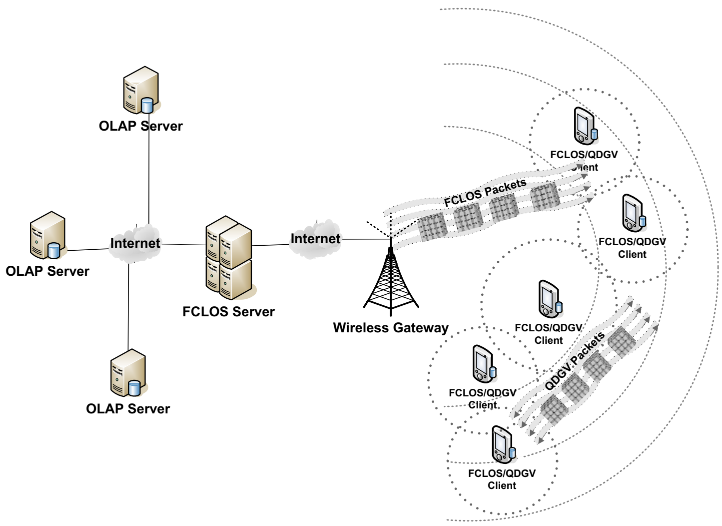

Figure 1.1: mOLAP architecture based on FCLOS and QDGV

sively rely on a server for query answering, by additionally incorporating ad hoc connections. Unfortunately, key ideas from the client-server mOLAP domain tackled with *FCLOS* cannot be directly applied in mobile ad hoc networks, primarily due to the shared wireless medium (no exclusive wireless channels), the fact that query propagation has to be tackled too (no central facilities), and that intelligent scheduling cannot provide any substantial optimization (no accumulated queries in one node). *QDGV* is a simple and adaptive *query propagation* and *data delivery* protocol. *QDGV* is by no means an all purpose data dissemination protocol, neither in regard to the running application, nor in regard to the network topology. It is explicitly designed for ad hoc mOLAP querying, under realistic application domain assumptions. *QDGV* does not employ any cross-layer technique and is completely independent of the underlying MAC and ad hoc routing protocol.

Figure 1.1 depicts a mOLAP architecture based on *FCLOS* for client-server transmission of multidimensional cubes and on *QDGV* for ad hoc exchange of data cubes between mobile clients. *FCLOS* and *QDGV* can be used simultaneously or completely independently.

1.3 Thesis Overview

The rest of this document is organized as follows:

Chapter 2 delivers a brief description of multidimensional databases, their key characteristics and their application areas. Given the breadth of the area, the description focuses on the most thesis relevant aspects. We focus on issues of query derivability in multidimensional databases and illustrate the role of aggregation lattices.

Chapter 3 describes the second thesis relevant research area, which is data management in wireless, mobile environments. It serves, like Chapter 2, as a general introduction to the tackled research problem. After a short presentation of relevant wireless communication technologies and the inherent limitations of the domain, we focus primarily on data management issues in wireless information broadcast, and secondarily on research issues concerning mobile databases in general.

Related work in the area of data broadcast is the topic of Chapter 4. The presentation involves general data broadcast systems, as well as systems dealing exclusively with database applications.

In Chapter 5, we present related work in the area of mobile data warehousing. Starting with pioneer approaches in distributed DWs, mainly in wired networks, the discussion involves the wireless aspect by referring to mobile DWs. The chapter ends with a thorough analysis and evaluation of approaches directly comparable to the *FCLOS* architecture.

Chapter 6 provides a detailed presentation of the *FCLOS* architecture, including the underlying data model as well as the server and client model. A formal description of the *FCLOS* scheduling component is followed by an extensive experimental evaluation of the proposed system. The evaluation is performed against numerous datasets, both synthetic and real, workloads, and network environments.

In Chapter 7, an analytical framework for computation of sub-cube derivability probabilities is introduced. This framework has a twofold purpose. On the one hand, it facilitates the suitability evaluation of different aggregation lattices (including or not hierarchical levels) for the mOLAP domain. On the other hand, it justifies the design choices of the *FCLOS* scheduler and its superiority against its competitors. The chapter concludes with an experimental evaluation, which confirms the validity of the analytical framework.

Chapter 8 presents an extension of the *FCLOS* architecture, which incorporates advanced data cube physical structures. It introduces the *m-Dwarf*, a compressed data cube physical structure, which has no loss of semantic information and is explicitly designed for mobile applications. The accompanied experimental evaluation demonstrates a substantial reduction of generated traffic due to the achieved compression.

In Chapter 9, we describe an entirely different aspect of the research problem. We deal with a secondary problem of the thesis, which is the exchange of multidimensional data in wireless ad hoc networks. New challenges both in comparison to querying of relational data in ad hoc networks and respective multidimensional querying in infrastructure based wireless networks are outlined. We introduce *QDGV*, a simple query propagation and data dissemination protocol, exclusively tailored for ad hoc mOLAP. Extensive performance evaluation validates the efficiency and robustness of *QDGV*.

Finally, Chapter 10 concludes the thesis and illustrates its main contributions. It additionally discusses issues yet to be addressed in the mOLAP domain.