Part IV

Conclusion

Chapter 10

Conclusion and Future Work

This chapter summarizes the contribution of this thesis and provides an outlook on future work in the area of mOLAP.

10.1 Overview

The research field of mOLAP encompasses all necessary technologies for mobile information systems that enable multidimensional data access to users carrying a mobile device.

This work argues that efficient and robust mOLAP cannot be provided by general data broadcast, mainly because such systems ignore data semantics and OLAP end user behavior, while assuming thin clients. Moreover, multidimensional data cubes are items order of magnitude bigger than the ones assumed by such systems. We show that mOLAP requires an entirely different architecture, which considers and exploits the characteristics of multidimensional data and the behavior of OLAP end users.

In this context, this dissertation presents *FCLOS*, a mobile information system explicitly designed for mOLAP. *FCLOS* is a *scalable* and *self-adaptive* system. The performance of *FCLOS* was evaluated both analytically and experimentally and outperformed its competitors on all relevant criteria, exhibiting optimization of more than 65%.

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, when the incoming load is relatively low, *FCLOS* exhibits the advantage of on-demand systems, while when the load increases, due to the subsumption exploitation, *FCLOS* exhibits the advantage of push-based systems, scalability.

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. This introduces an indirect tradeoff, which can be controlled in several ways.

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

We identified the exploitation degree of sub-cube derivability as the critical performance parameter. Motivated by this fact, we introduced an analytical framework that facilitates the computation of sub-cube derivability probabilities in both lattices. This framework, besides providing the basis for a general evaluation, focuses on the specific domain of mOLAP dissemination systems. The analytical framework revealed that sub-cube derivability is optimally exploited in the case of *DCL*. Experimental results for state of the art mOLAP systems confirmed that both server and mobile clients benefit from *DCL* mapping.

Scheduling decisions are completely separated from the size of the objects being handled. *FCLOS* considers dimensionality instead of size, which is *independent of the data cube 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.

We introduced the *m-Dwarf*, a highly compressed data cube physical structure, explicitly designed for mOLAP. The *m-Dwarf* structure can be seamlessly integrated into the *FCLOS* architecture. The structure's storage savings come at the expense of missing aggregated values, which are eventually undertaken by the client.

Finally, we presented the case of mOLAP in ad hoc networks and introduced *QDGV*, a query propagation and data dissemination 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 environment. 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.

10.2 Comparison of mOLAP Architectures

In Section 5.3.5, we compare state of the art mOLAP systems according to the specified criteria. Table 10.1 is an extended version of Table 5.1, which includes *FCLOS* as well. It summarizes the achievements of our proposal. Again, the criteria are weighed using the symbols of Table 4.1.

To facilitate the analysis, Fig. 10.1 depicts the mean query access time for all approaches examined in this thesis.

Table 10.1: Evaluation of mOLAP architectures

	P to P	STOBS	SBS	DV-ES	FCLOS
Performance	- -	0	0	0	++
Offline functionality	- -	0	0	0	++
Scalability	- -	0	0	0	++
Self adaptiveness	-	- -	- -	- -	++
Load balancing	- -	0	0	0	+
Complexity	++	-	- -	- -	- -
Fairness	+	-	+	+	- -
Maintenance overhead	++	-	- -	- -	+
Application generality	++	- -	- -	- -	- -
Extendability/Modularity	- -	-	-	- -	+

Performance

Extensive experimental evaluation demonstrated the performance gain of *FCLOS* on all criteria: query access time, energy consumption overhead and generated traffic. Sections 6.6, 8.5 deliver an insightful analysis.

Offline Functionality

Offline functionality is directly related to the available data, which in turn is strongly affected by the query mapping scheme. In Chapter 7, we analytically substantiate the choice of *DCL* query mapping scheme and the superiority of the *FCLOS* in this respect. Figure 6.10 reveals that *FCLOS* rarely has to resort to a server connection, thus exhibiting enhanced offline functionality.

Scalability

This measures the extent to which an architecture can handle growing number of incoming requests. Figures 10.1, 6.22, 6.16 prove *FCLOS*' scalability.

Self Adaptiveness

Broadcast systems are usually designed under specific workload assumptions. However, workloads are not static. This criterion evaluates the degree in which an architecture can adapt itself to workload changes. Figures 6.21, 6.22, 6.23 demonstrate *FCLOS'* independence of query distribution and incoming load.

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

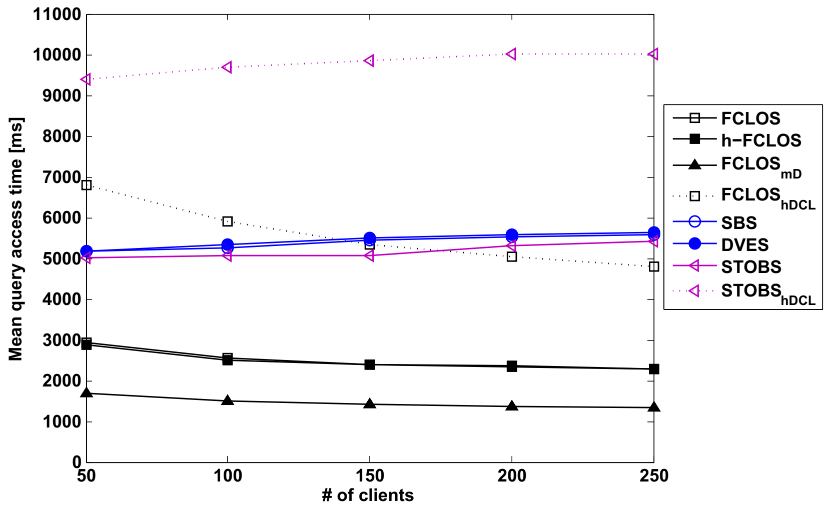

Figure 10.1: Mean query access time (T_{all}) for all approaches examined in the thesis

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. 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.

Complexity

FCLOS's complexity is rather higher than the one of its competitors. This is justified by the additional overhead of searching for broadcast clusters, as described in Section 6.4.

Fairness

Figure 6.17 revealed that *FCLOS* is outperformed by its competitors, as far as the stretch is concerned. *FCLOS* fundamental design principle is to serve as many clients as possible with each broadcast. Inevitably, queries targeting smaller datasets frequently have to deal with bigger datasets. This incurs increased stretch.

Maintenance Overhead

As seen in Fig. 6.19, *FCLOS* does not depend on runtime configuration parameters, unlike its competitors, which have to control the derivation flexibility with the a-optimizer.

Application Generality

FCLOS is an architecture, explicitly designed for mOLAP. It is practically of little use for other application domains.

Extendability/Modularity

As seen in Chapter 8, the modular architecture of *FCLOS* enables a seamless integration of any data cube physical structure.

10.3 Future Work

Naturally, not all issues were addressed by this dissertation.

The performance of *FCLOS* was experimentally and analytically evaluated. Naturally, in order to gain a more profound understanding of the system, it would be very interesting to evaluate its performance in real application scenarios using real hardware.

Another aspect not addressed yet is node heterogeneity. *FCLOS* assumes that all clients are capable of performing local processing and this with the same capacity, having adequate energy resources. It is worth investigating scenarios in which a client portion is not capable of local processing and thus cannot be served by ancestor sub-cubes.

The *FCLOS* architecture assumes one downlink channel for every disseminated cube. In case of more than one disseminated cube, *FCLOS* can use more than one downlink channel in order to serve clients. Given the huge amount of related work in the area of scheduling in multiple channels, it would be interesting to examine how existing work can be integrated in our architecture in order to further optimize scheduling.

The issue of preemption is indirectly addressed in the runtime connection handling of our ad hoc protocol, but no preemption is used by *FCLOS* in the infrastructure scenarios. In this context, it would be interesting to investigate if preemption techniques can be of benefit for infrastructure mOLAP.

Our system exhibits great scalability as far as client population and incoming load is concerned. However, scalability in terms of the size of transmitted cubes is a different story. Although we believe that it is unrealistic to assume tera byte DWs in mOLAP, it is worthwhile exploring the upper bound that *FCLOS* is able to handle.

The introduced analytical framework considers the dimensionality and number of hierarchical levels of a *MDDB*. Future work should focus on extending the framework by considering additional metadata information, like cardinality of each hierarchical level and database size. The intuition behind this is that in applications other from mOLAP, where *MDDBs* are much bigger, it might be beneficial to use the *hDCL* instead.

As underlined in Chapter 8, the compression achieved by the *m-Dwarf* comes at the cost of poor indexing. Although this is an intended compromise in order to minimize the size of the transmitted structure, it is definitely worthwhile investigating what kind of indexing can be applied on the *m-Dwarf* upon reception of the structure in order to enhance subsequent query performance. Thus, the conversion to some other physical structure can be avoided.

Finally, it would be interesting to compare mOLAP architectures such as *FCLOS*, *SBS* and *STOBS*, which transmit entire sub-cubes, with more fine-grained techniques, which transmit chunks instead, such as the one in [92].

Future work concerning ad hoc mOLAP and *QDGV* is explicitly presented in Section 9.7. The integration of *QDGV* in the *FCLOS* architecture may be a next step for future work.