

# Semantic Caching in a Mobile Environment

Say Ying Lim

*Monash University, Australia*

S

## INTRODUCTION

Mobile computing environments enable the database servers to disseminate data via wireless channels to multiple mobile clients (Chung & Kim, 2001). It has increased popularity with the emerging trend of wireless network and usage of handheld devices, such as PDAs and other portable electronic devices. The typical nature of a mobile environment would include low bandwidth and low reliability of wireless channels, which causes frequent disconnection to the mobile users. Hence due to the constraints of the nature of mobile environment it is important to enhance the performance of the query processing, as well as improve the availability of querying particular data items especially during disconnection (Imelinkski & Korth, 1996; Malladi & Davis, 2002). Often, mobile devices are associated with low memory storage and low power computation and with a limited power supply (Myers & Beigl, 2003). Hence, it is important to help mobile clients to save the usage of its battery.

By introducing data caching into the mobile environment, it is believed to help improve data availability in case of disconnection by retrieving data that has been previously cached in the local memory and to be able to save power by having a lower data transmission. Generally, in data caching, it means the data is cached in the memory storage of the mobile device, and whenever the mobile users want to issue a query, it will first search its cache and if there exists a valid copy in the cache it returns the results immediately. Otherwise the mobile users would attempt to obtain the data from the server either using the server or broadcast strategy. Caching has emerged as a fundamental technique, especially in distributed systems, as it not only helps reduce communication costs but also off loads shared database servers.

In this article, we describe the use of caching, which allows coping with the characteristic of the mobile environment. We concentrate particularly on semantic caching, which is basically a type of caching strategy that is content-based reasoning ability with the ability to—in addition of caching query results—remember the queries that generated these results. Semantic caching provides accurate, semantic description of the content of the cache.

## BACKGROUND

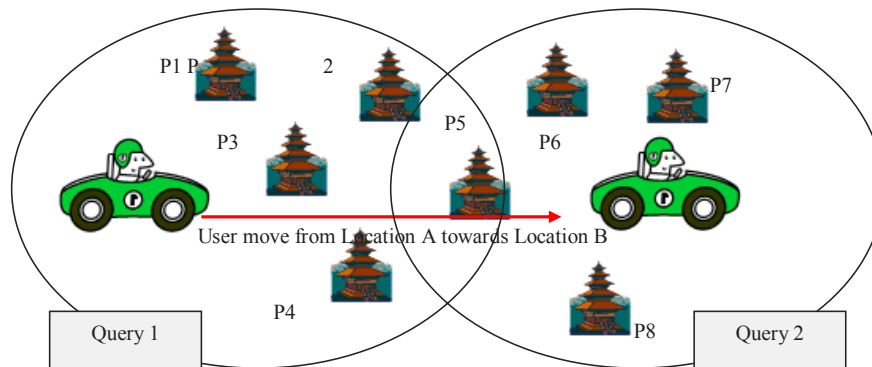
The effect of having the ability to cache data is of great importance, especially in the mobile computing environment than in other computing environments. This is due to the reason that contacting the remote servers for data is expensive in the wireless environment and, with the vulnerability to frequent disconnection, can further increase the communication costs (Leong & Si, 1997). There are many different types of caching strategies that serve the purpose to improve query response time and to reduce contention on narrow bandwidth (Zheng, Lee, & Lee, 2004). Caching mechanisms need to retain the frequently accessed data locally in the mobile device storage to be able to allow users to access server database queries at least partially in cases of disconnections. Hence, the more effective the caching mechanism is in keeping the frequently accessed data will result in more queries that can be served during disconnection.

Due to limitations such as cache space, cache replacement and cache granularity, as well as cache coherence, are the three main issues that characterize caching mechanism. In traditional cache replacement, the most important factor affecting cache performance is the access probability. This refers to replacing the data with the least access probability to free up more cache space for the new data. There is a large variety of caching replacement policies and most of them utilize access probability as the primary factor in determining which data items are to be replaced.

Cache granularity relates to determining a physical form of cached data items. It appears to be one of the key issues in caching management systems. There are three different levels of caching granularities in object-oriented databases, which includes: (a) attribute caching, (b) object caching and (c) hybrid caching (Chan, Si, & Leong, 1998). Attribute caching refers to frequently accessed attributes that are stored in the client's local storage. As for object caching, instead of the attribute itself being cache, the object is cached. In attribute caching, it creates undesirable overheads due to the large number of independent cache attributes. Thus, hybrid caching, which appears to be a better approach, comprises of the combinations of both granularities.

Cache coherence—or known as invalidation strategy—involves cache invalidation and update schemes to invalidate

Figure 1. Overlapping results from two queries issued



and update out-dated or non valid cached items (Chan, Si, & Leong, 1998; Cao, 2003). After a certain period, a cached data may appear as no longer valid and therefore mobile users should obtain a newer cache before retrieving the data (Xu, Tang, & Lee, 2003). There are several techniques that have been proposed to overcome this issue. These include (a) stateful server, (b) stateless server, (Barbara & Imielinski, 1994) and (c) leases file caching mechanism (Lee, Leong, & Si, 2001). Stateful server refers to the server having an obligation to its clients, which means the server has the responsibility in notifying the users about changes, if there are any. In contrast, stateless server refers to the server not aware for its clients, whereby the server broadcasts a report that contains the updated item either asynchronously or synchronously. The leases files mechanism, which is also known as lazy invalidation approach, assigns each mobile user to be responsible for invalidating its cached items.

Consequently, a good caching management strategy is needed to deal with the critical caching issues, such as caching replacement, caching granularity and caching coherence.

## SEMANTIC CACHING

A better way of query processing specifically for use in a mobile environment is by allowing the users to specify precisely what data items are missing from its local storage to server the query. This could be achieved by having the previously evaluated query results being cached (Dar et al., 1996; Roussopoulos, 1991).

### Using Semantic Caching in a Mobile Environment

A semantic cache is defined as consisting of a set of distinct semantic segments, which can be decomposed into separate components or come together as a whole of the query

results. A semantic segment  $S$  can be specified by having  $\langle SR, SA, SP, SC \rangle$  whereby  $SR$  and  $SA$  define the base relation of relation and attributes in the creation of the semantic segment respectively.  $SP$  is to indicate the criteria that  $S$  satisfies, and  $SC$  indicates the actual content of  $S$ , which is represented by pages. (Ren, Dunham, & Kumar, 2003)

Semantic caching stores semantic descriptions and associated answers of the previous queries in the mobile client (Dar et al., 1996). The main feature of semantic caching is the content-based reasoning ability as well as the fact that only the required data, as opposed to a file or pages of data, is transmitted over the wireless channel.

When a new query exists, the mobile client can determine whether should it be totally answered by how much can it be answered and what data are missing. With these abilities, the wireless traffic can be greatly reduced because only the needed data are transferred. This helps with disconnection too, since total or partial results may be obtained even when the server is unreachable (Lee, Heong, & Si, 1999). As a result, if a query can be partially answered from the cache, the volume of missing data requested from the server as well as the wireless bandwidth consumed can be reduced. And if the query could be answered completely based on the cache, then no communication between the client and the server is required at all. This ability is of particular significance during disconnection, which is the main constraint the mobile environment is currently facing. This also leads to reduction of overhead due to redundant computation as the amount of data transferred over the wireless channel can be substantially reduced.

**Example 1:** Suppose a mobile user who is traveling from one location to another location suddenly wished to find a nearby rest place. So the user issue a query while he is in Location A and the server returns the nearest rest place which is  $P1, P2, P3, P4, P5$ . But the user is not satisfied with the results. So he re-issued another query while he is moving

5 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: [www.igi-global.com/chapter/semantic-caching-mobile-environment/17185](http://www.igi-global.com/chapter/semantic-caching-mobile-environment/17185)

## Related Content

---

### An Ontology-Based Approach for Mobile Agents' Context Awareness

N. Amara-Hachmi and A. El Fallah-Seghrouchni (2007). *Encyclopedia of Mobile Computing and Commerce* (pp. 717-723).

[www.irma-international.org/chapter/ontology-based-approach-mobile-agents/17162](http://www.irma-international.org/chapter/ontology-based-approach-mobile-agents/17162)

### Comb Filters Characteristics and Current Applications

Miriam Guadalupe Cruz-Jimenez, David Ernesto Troncoso Romero and Gordana Jovanovic Dolecek (2019). *Advanced Methodologies and Technologies in Network Architecture, Mobile Computing, and Data Analytics* (pp. 791-804).

[www.irma-international.org/chapter/comb-filters-characteristics-and-current-applications/214661](http://www.irma-international.org/chapter/comb-filters-characteristics-and-current-applications/214661)

### Mobile Data Offloading Using Opportunistic Communication and AP Deployment: A Case Study

Sanjit Kumar Dash, Sasmita Mishra and Jibitesh Mishra (2017). *International Journal of Mobile Computing and Multimedia Communications* (pp. 66-84).

[www.irma-international.org/article/mobile-data-offloading-using-opportunistic-communication-and-ap-deployment/193260](http://www.irma-international.org/article/mobile-data-offloading-using-opportunistic-communication-and-ap-deployment/193260)

### Improving Throughput of Starved TCP Flow by Sidestepping Bottleneck Nodes Using Concurrent Transmission

Rajesh Verma, Arun Prakash, Rajeev Tripathi and Neeraj Tyagi (2010). *International Journal of Mobile Computing and Multimedia Communications* (pp. 68-83).

[www.irma-international.org/article/improving-throughput-starved-tcp-flow/40981](http://www.irma-international.org/article/improving-throughput-starved-tcp-flow/40981)

### Intelligent User Interfaces for Mobile Computing

Michael J. O'Grady and Gregory M.P. O'Hare (2008). *Handbook of Research on User Interface Design and Evaluation for Mobile Technology* (pp. 318-329).

[www.irma-international.org/chapter/intelligent-user-interfaces-mobile-computing/21839](http://www.irma-international.org/chapter/intelligent-user-interfaces-mobile-computing/21839)