ABMMCCS: APPLICATION BASED MULTI-LEVEL MOBILE CACHE CONSISTENCY SCHEME

Doha Elsharief[1], Hamidah Ibrahim[2], Ali Mamat[3], Mohamed Othman[4]

[1]*Universiti Putra Malaysia (UPM), Malaysia*, dohayagoub@yahoo.com
[2, 3, 4]*Universiti Putra Malaysia (UPM), Malaysia, [2]hamidah, [3]ali,[4]mothman@fsktm.upm.edu.my*

ABSTRACT. Maintaining cache consistency in mobile computing system is a critical issue due to the inheritance limitations in mobile environment such as limited network bandwidth and mobile device energy power. Most of the existing schemes maintaining mobile cache consistency support only one level of consistency that is either strict or weak which is not suitable all the time, as various mobile applications systems have different consistency requirements on their data. Also majority of the schemes restrict the using of cached data for reading only which is limits the functionality of the caching system. In this paper, a new scheme is proposed to maintain the mobile cache consistency in a single cell wireless network called Application Based Multi-Level Mobile Cache Consistency Scheme (ABMMCCS). The main idea in ABMMCCS is to be suitable to various real mobile application systems, by supporting multiple levels of consistency based on the application requirements, while saving the mobile client energy power and reducing the consumption of the network bandwidth. The initial evaluation results show that, ABMMCCM reduces the number of uplink messages issued from the mobile client, which is assist in saving the mobile client energy and better utilizing the limited network bandwidth.

Keywords: mobile cache consistency, stateful approach, multi-level cache consistency scheme

INTRODUCTION

Recent expansion in wireless networking and the exponential development in mobile devices have emerged a new paradigm in distributed systems is a mobile computing. In this paradigm, users who carry portable devices have access to the desired information services and access data reside any where through a shared infrastructure regardless of their physical location or movement behavior. The mobile computing environments are suffering from many problems among which is the constricted bandwidth, limited resources on mobile devices, and disconnectivity (Safa *et al.*, 2008). Caching frequently accessed data in the mobile client has a vital role in improving the performance of the various mobile computing systems by increasing the availability of data in the presence of disconnectivity, reducing the data retrieval from the original server, relieving the bandwidth consumption, and reducing the latency in data access (Huang *et al.,* 2007). The replicas of the same data object distributed among multiple mobile clients need a consistency between them and with the original data resides in the original source. Due to the mobile environment inherited problems, maintaining cache consistency in a mobile environment is a complicated process.

In the literature, three levels of mobile cache consistency are introduced: *strong consistency* where the cached data item in the mobile unit is up to date with the original data at the source, *delta consistency* in which the cached data in the mobile client is never out of date by more than a specific time with the original data at the server, and *weak consistency* level in which the data item cached at the mobile client is a copy of the previous version of the source data item at the source (Cao *et al.*, 2007). Most of the schemes in maintaining mobile cache consistency support one level of consistency for example strict (Barbar and Imielifiski, 1995; Chan *et al.*, 2005; Chuang and Chiu, 2008; J.C. and K, 2009; Madhukar *et al.*, 2009; Pamila and Thanushkodi, 2009; Safa *et al.*, 2008; Wang *et al.*, 2010; Xu *et al.*, 2002; Yi *et al.*, 2007), or weak such as (Chan *et al.*, 2005; Kumar *et al.*; 2009;Yuen *et al.*, 2000; Zhijun Wang *et al.*, 2004). Applying only one level of consistency on the cached data items is not appropriate all the time, as some applications allow some degree of weak consistency on the cached data and some critical cached data items have to be up to date with original data in the source. However most of the schemes restrict the cached data items in mobile units for read only which limits the caching system functionality.

Vora, (2005) proposed a stateful method called Multi-level Cache Consistency Protocol (MCCP). The protocol makes distinction between strict and weak consistency levels, and the mobile user able to issue update operations on its own cached data. There are some drawbacks on this scheme such as: a)the different levels of consistency it supports depend on the mobile client interest, which is not a good idea, as in real systems the data consistency depend on the application requirement, b)the process to determine the consistency requirements incurs overhead to the mobile client, it consumes the client's energy and bandwidth of wireless network especially the uplink channel, c)single data item may have many consistency requirements (depending on the number of the cached clients), which represent overhead on the base server to maintain its consistency. We proposed a scheme called Application Based Multi-level Mobile Cache Consistency Scheme (ABMMCCS) based on Multi-level Cache consistency Protocol (MCCP). ABMMCCS scheme preserves the advantages of MCCP and enhances its drawbacks.

The rest of the paper is organized as follows. Section 2 describes the proposed multi-level mobile cache consistency scheme ABMMCCS, its system architecture and main features, while the details design of the proposed scheme ABMMCCS is presented in Section3. Initial evaluation of the proposed scheme is reported in Section 4. The conclusion is offered in Section 5.

PROPOSED MULTI-LEVEL MOBILE CACHE CONSISTENCY SCHEME

System Architecture of ABMMCCS

Figure 1, shows a typical mobile computing environment. It consists of a number of Mobile Host (MHs)/Mobile Clients (MCs) and powerful fixed hosts that are connected through a wired network. Some of the fixed hosts, called Base Servers/Base Stations (BSs) are equipped with a wireless interface to communicate with mobile clients, which are located within a coverage area called a wireless cell (Madhukar *et al.*, 2009; Safa *et al.*, 2008). The client is free to move within its cell or out of cell range (handoff). Through the BS the mobile client is able to access to other networks by forwarding its functions. Also BS performs various functions such as handoff, mobility management(Vora, 2005).

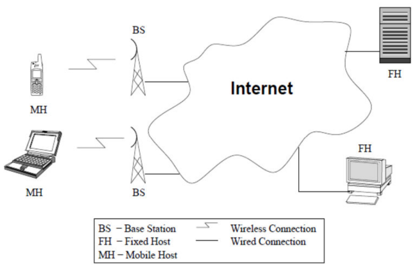

Figure 1. Mobile Computing System Architecture (Vora, 2005)

Main Features of ABMMCCS

ABBMMCCS scheme supports multiple levels of consistency (strict/different levels of weak) based on the application requirements. This feature enables ABMMCCS to be suitable in maintaining the cache consistency of variance mobile applications, also in ABMMCCS; each single data item has a single consistency requirement entry, which is decreases, the overhead of the base server in maintaining the consistency of the item. However; in ABMMCCS the mobile client is released from determining the consistency requirements of its cached data, which reduces the overhead of the mobile client, save its power energy, and better utilized the narrow uplink channel.

ABMMCCS SYSTEM MODEL

The components of ABMMCCS system model are: the Application System Developer (ASD), the Base Server (BS), and the Mobile Client (MC).

ASD Working Model

In ABMMCCS, the mission of ASD is to design data consistency requirements**.** In ABMMCCS, the consistency requirements of cached data items are based on the mobile application requirements. ASD specifies the granularity of data at which they would like to maintain application consistency, creates the Consistency Table (CT), and storing it on the database server. Each data item has a single record in CT with the following form:

$$(Did, P, £T)$$

Where *Did* is the data item identifier, *P* is the predicate condition on the data item (when violated the data item become inconsistent), and *£T* a time point after it the cached data item is no longer consistent, or a maximum delay time allowed to server before propagating the update to the cached client. If *P and time delay £T are null,* this indicates strict consistency; if one or both of *P and £T is null* this *indicates* a level of weak consistency.

BS Working Model

In ABMMCCS, the BS has three main functions; it controls the access to the data items, maintains the data consistency, and propagates updates to the mobile clients.

Data Access Control

In ABMMCCS BS controls the access to the data items based on its consistency requirements. The concept of lock is used as in MCCP; four types of locks are available on the data items. They originate from the data items consistency requirements and the operations available on the data items (Read/Write). There are two types of strict locks namely; *Strict Read Lock (SRL)* and *Strict Write Lock (SWL)* and two weak locks *Weak Read Lock (WRL)* and *Weak Write Lock (WWL)*. Table 1 summarizes the different lock types defined on a data item. The vertical axis represents the data item consistency (Strict /Weak), while the type of operations available on the data item (Read/Write) is represented in the horizontal axis. Any cache lock request message issued by client, BS checks its compatibility with the item consistency requirements before accepts or rejects the request. Table 2 summarizes the compatibility of the lock types with the required operations. The horizontal axis represents the lock type currently exists on a data item, while the vertical axis represents the required operation from client.

Table 1. Lock Types Available to Client on the Data Items

Operation Type	Strict Consistency	Weak Consistency
Read	SRL	WRL
Write	SWL	WWL

Table 2. Compatibility of Lock Types with the Client Operations

Operation Required	**None**	SRL	SWL	WRL	WWL
Read	✓	✓	×	✓	✓
Write	✓	×	×	✓	×

✓: Compatible ×: Incompatible

Maintaining Data Consistency and Updates Propagation

Based on the consistency requirements of the data item, BS determines the type and time of propagating the received updates on the data item to the affected clients. Four levels of consistency are discriminated based on the values of *Predicate condition (P)* and *time delay* (*£T*) namely; *Strict Updates Strict Notify (SUSN), Weak Updates Strict Notify (WUSN), Strict Update Weak Notify (SUWN),* and *Weak Updates Weak Notify (WUWN).* Table 3 summarizes the propagation of updates by BS based on the levels of consistency.

Table 3. Propagation of Updates Based on the Data Consistency Levels

Consistency Level	Immediate Propagation	Delay	All	Some
SUSN	Yes	No	Yes	No
WUSN	Yes	No	No	Yes
SUWN	No	Yes	Yes	No
WUWN	No	Yes	No	Yes

MC Working Model

The role of the MC in the model is to:

- Demand a required data item from the base server: the request message besides the required item contains the operation lock required on the item (read/write) and the lock period required.
- Send updates to the base server: The client holds a write lock on cached data item, sends each update directly to BS (the data item has weak consistency) or one update at the end of a lock period or if the client determines to release the item lock (strict consistency).

- Validate its cache: when the client received an update from BS, the mobile client sends an acknowledgment message to BS, and then validates its cache by committing or rolling back the conflicting operations with the received update.

INITIAL EVALUATION OF THE PROPOSED SCHEME

Compare ABMMCCS to MCCP with respect to the consumption of the MC energy power and the wireless network bandwidth, initially as a tentative result Figure 2 shows the possible number of messages transferred between MC and BS using ABMMCCS presented in (a) and MCCP in (b). It is depict from the figure that, using ABMMCCM to maintain

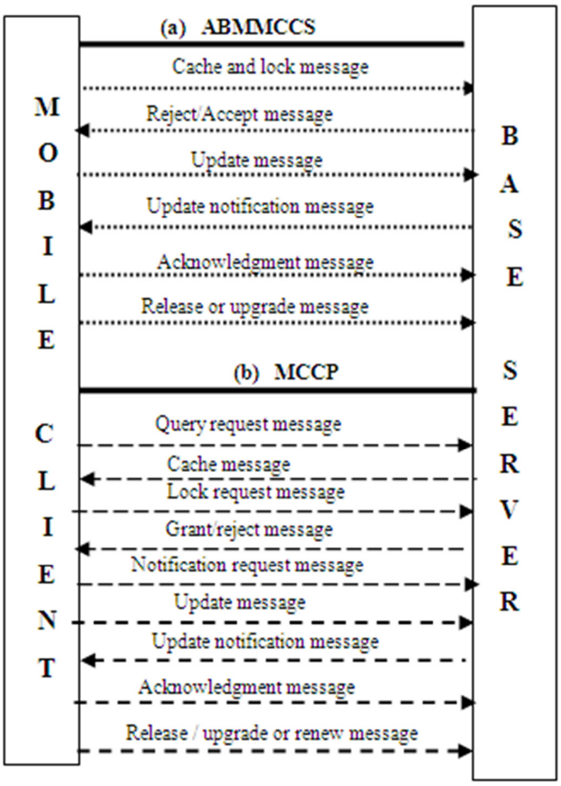

Figure 2. Caching Process - Messages Exchanges
(a) ABMMCCS (b) MCCP

the mobile cache consistency, the number of uplink messages issued from MC decrease with the rate approximately 33% compared to MCCP which is help in saving the client's energy(by saving the power consumed in transmitting and receiving messages) and reducing the consumption of the limited uplink channel bandwidth.

CONCLUSION

In this paper we proposed a new scheme called Application Based Multi-level Mobile Cache Consistency Scheme (ABMMCCS). ABMMCCS suitable to various mobile application systems, by supporting multiple levels of consistency based on the application requirements, as variance applications have different consistency requirements on their data. ABMMCCS is based on MCCP scheme (Vora, 2005). From the initial performance evaluation of ABMMCCS compared to MCCP, we could conclude that ABMMCCS supports multiple levels of consistency, while enhancing the consumption of mobile client limited energy power and the utilization of wireless network bandwidth. Simulation modeling and analytical study are to be considered in the future research.

REFERENCES

Barbar, D. andImielifiski, T. (1995). Sleepers and Workaholics: *Caching Strategies for Mobile Environments (Extended Version). VLDBJoumal, 4*, 567-602.

Cao, J. , Zhang, Y. , Cao, G., and Xie, L. (2007). *Data Consistency for Cooperative Caching in Mobile Environments*. *IEEE Computer Society, 40*(4), 60-66.

Chan, E. , Woo, L. H., and Yuen, J. C. H. (2005). *IAVI-UIR: An Efficient Caching Scheme for Mobile Computing Systems*, proceeding of the 27th Int. Conf. Information Technology Interfaces ITI, June 20-23, Cavtat, Croatia.

Chuang, P-J., and Chiu, Y.-S. (2008). *Constructing Efficient Cache Invalidation Schemes in Mobile Environments.* Proceeding of the Third International IEEE Conference on Signal-Image Technologies and Internet-Based System (SITIS 2007).

Kumar, C , Kiranmayi, K., Bhargava, A., and Mohanta, D. (2009). *Efficient Bandwidth Utilization in Mobile Environments for Cache Consistency Using Queried Data (CCUQD) Approach.* proceeding of the India Conference (INDICON).

Madhukar, A., Zyer, T. O., and Alhajj, R. (2009). Dynamic Cache Invalidation Scheme For Wireless Mobile Environments. Wireless Networks(2009) 15, 727–740.

Pamila, M. J. and Thanushkodi (2009). *Performance Analysis of Improved Cache Invalidation Scheme in Mobile Computing Environment*. International Journal of Computer Science and Network Security(IJCSNS), 9 (9).

Safa, H., Artail, H., and Nahhas, M. (2008). *Enhancing Cache Invalidation in Mobile Environments.* Proceeding of the International Conference on Mobile Technology, Applications & Systems 2008 (Mobility Conference), Ilan,Taiwan.

Vora, A. A. (2005). *Data Stashing Strategies for Disconnected and Partially Connected Mobile Environments*. Ph.D. Thesis, RMIT University.

Wang, J., Zhou, Z., and Liu, M. (2010). *An Improved Cache Invalidation Schema Based On Mobile Agent.* Proceeding of the Future Computer and Communication (ICFCC), Wuhan

Xu, J. , Lee, D.-L. , HU, Q., and LEE, W.-C. (2002). *Broadcasting Data*. In I.Stojmenovic (Ed.), *Handbook of Wireless Networks and Mobile Computing*: John Wiley & Sons.

Yi, S.-Y., Jung, S., and Suh, J. (2007). *Better Mobile Client's Cache Reusability and Data Access Time In A Wireless Broadcast Environment*. ScienceDirect- Data & Knowledg Engineering, 63 293–314.

Yuen, J. C. H., Chan, E., Lam, K.-Y., and Leung, H. W. (2000). *Cache Invalidation Scheme for Mobile Computing Systems with Real-time Data*. ACMsigmod Record, 29(4), 34 – 39.

Zhijun Wang, Das, S. K., Che, H., and Kumar, M. (2004). *A Scalable Asynchronous Cache Consistency Scheme (SACCS) for Mobile Environments*. IEEE Transactions on Parallel and Distributed Systems, 15(11).