



CS Cache Engine: Data Access Accelerator for Location-Based Services in Mobile Environments



Ken C. K. Lee Wang-Chien Lee Baihua Zheng Jianliang Xu
CSE, Penn State University Singapore Management University HK Baptist University

Abstract: Client-side data caching plays an important role of enhancing data availability and improving the query response time in wireless and mobile environments. In this presentation, we present *CS Cache Engine* suitable for *Location-Based Services*, one of the killer applications for mobile and pervasive computing environments. The CS Cache Engine implements *Complementary Space Caching* model that preserves a global view of the database by capturing those objects not in the cache but in the server as Complementary Region (CRs) in addition to cached data objects in the cache. With the CS Cache Engine implementing CS caching, various kinds of location-based queries can be supported and client assertiveness on their own answered queries is enhanced, greatly reducing unnecessary requests over the wireless channel.

Introduction

Location-dependent query: (example)

- Where is the *nearest* Italian restaurant?
- Where are available hotels within a mile from here?



Pushing technology:

- ✓ Wireless communications (e.g., WiFi, Bluetooth),
- ✓ Handheld devices (e.g., PDA), and
- ✓ Positioning systems (e.g., GPS)

Constraints of mobile environment:

- ❖ Limited bandwidth and weak connectivity, and
- ❖ Short battery life

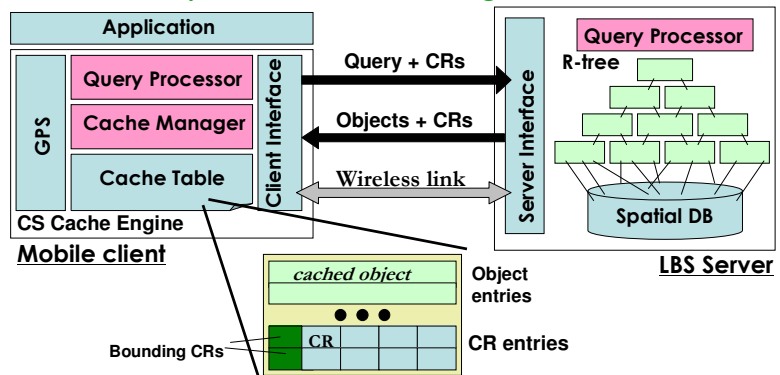
Our solution – Complementary Space (CS) Caching (at client):

- ❖ Support multiple kinds of location-dependent queries
- ❖ Maximize the cache answerability

System Architecture

Client/Server Model:

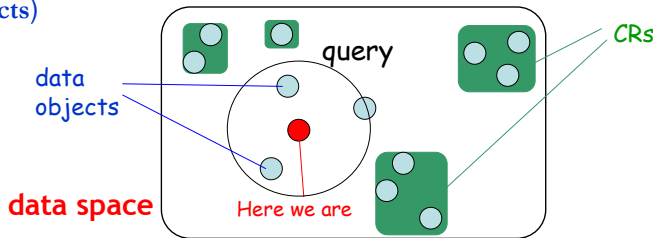
- ❖ Server: Query Processor, R-tree, and Spatial Database
- ❖ Client: Query Processor, Cache Manager, and Cache Table



CS Caching Model

What to cache? A global view of a dataset:

- ❖ Data Objects
- ❖ Complementary Regions (CRs) (abstractions of non-cached objects)



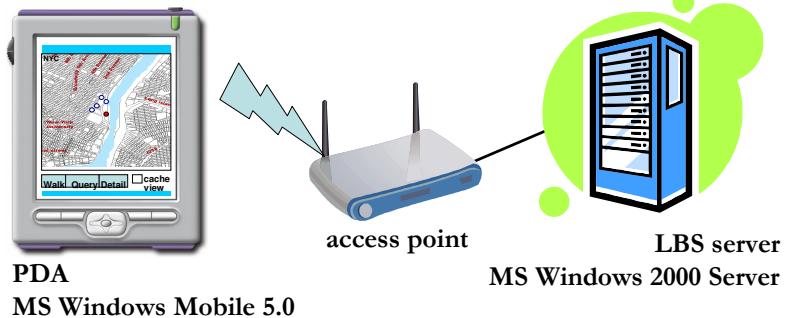
Technical challenges

- ❖ Limited cache space → space allocation for objects and CRs
- ❖ Query processing and cache replacement mechanisms

System Prototype

TouristGuide

- Searching facilities in New York City



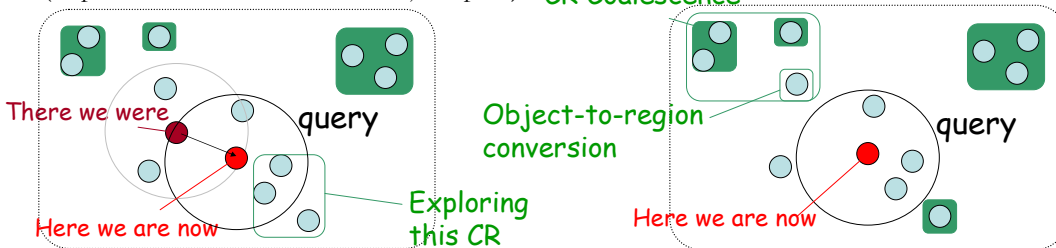
Query Processing & Cache Replacement

Query processing:

- Object and CR lookup
- Remainder query formulations (Explore CRs as if zoom-in the object space)

Cache replacement (FAR policy):

- Convert objects into CRs;
 - CR Coalescence (as if zoom-out the object space)
- CR Coalescence**



References

- K. C. K. Lee, W.-C. Lee, B. Zheng, and J. Xu, *Caching Complementary Space for Location-Based Services*, In **EDBT**, 2006
- K. C. K. Lee, W.-C. Lee, J. Winter, B. Zheng, and J. Xu, *CS Cache Engine: Data Access Accelerator for Location-Based Services in Mobile Environments*, In **SIGMOD Demo**, 2006
- Q. Ren, M. H. Dunham. *Using Semantic Caching to Manage Location Dependent Data in Mobile Computing*. In **Mobicom**, 2000