Research Challenges in Location-Enabled M-Services

Christian S. Jensen
Department of Computer Science
Aalborg University
Denmark
www.cs.auc.dk/˜csj

Abstract

Rapid, sustained advances in key computing hardware technologies combine to enable a new class of computing services that aim to meet needs of mobile users. These ubiquitous and intelligent services adapt to each user's particular preferences and current circumstances—they are personalized. The services exploit data available from multiple sources, including data on past interactions with the users, data accessible via the Internet, and data obtained from sensors. The user's geographical location is particularly central to these services.

We outline some of the research challenges that aim to meet the computing needs of such services. In particular, focus is on update and query processing in the context of geo-referenced data, where certain challenges related to the data representation, indexing, and precomputation are described.

1. Introduction

Several trends in hardware technologies combine to provide the enabling foundation for m-services. These trends include continued advances in *miniaturization* of electronics technologies, *display devices*, and *wireless communications*. Other trends are the improved *performance* of general computing technologies and the general improvement in the *performance/price ratio* of electronics hardware. Perhaps most importantly, *positioning technologies* such as GPS (global positioning system) are becoming increasingly accurate: the accuracy of GPS is slated to soon reach 5 meters.

It is expected that the coming years will witness very large quantities of wirelessly on-line, i.e., Internet-worked, objects that are location-enabled and capable of movement to varying degrees. Some predict that each of us will soon have approximately 100 on-line objects.

Example objects include consumers using WAP-enabled mobile-phone terminals and personal digital assistants, tourists carrying on-line and position-aware "cameras" and "wrist watches," vehicles with computing and navigation equipment, etc.

These developments pave the way to a range of qualitatively new types of Internet-based services. These types of services—which either make little sense or are of limited interest in the traditional context of fixed-location, PC- or workstation-based computing—include the following:

- traffic coordination, management, and way-finding,
- location-aware advertising,
- integrated information services, e.g., tourist services,
- safety-related services, and
- location-based games that merge virtual and physical spaces.

The area of location-based games offers good examples of services where there is a need to track the positions of the mobile users. In the recently released BotFighters game, by the Swedish company It's Alive, players get points for finding and "shooting" other players via their mobile phones (using SMS messages or WAP). Only players close by can be shot. To enable the game, players can request the positions of other nearby players. In such mixed-reality games, the real physical world becomes the backdrop of the game, instead of the world created on the limited displays of wireless devices [6]. These games are expected to generate very large revenues in the years to come. Datamonitor, a market research company, estimates that 200 million mobile phone users in western Europe and the United States will play Web games via handsets, generating $6 billion in revenues [13].

One single, *generic scenario* may be envisioned for such location-enabled m-services. Moving objects use services that involve location information. The objects disclose their positional information (position, speed, velocity, etc.) to the services, which in turn use this and other information to provide specific functionality.

Each service maintains a log, a so-called click-stream, of the requests made to it, and data derived from the click-stream is combined with other customer data. This data is used for analyzing the user interaction with the service and is used for mass-customization of the service, so that each user receives a service customized to the user's specific preferences and needs and current situation, e.g., dynamic, user-specific web-page content. In addition, the accumulated data is used for delayed modification of the services provided, and for longer-term strategic decision making.

The integration of location information into this scenario has received little attention and offers a number of fundamental challenges. Common to these challenges is the theme of extending techniques that work well for static data to support dynamic, continuously evolving data. Another common theme is to extend techniques, which have previously been subjected only to "simple" data, to support more complex types of data and queries.

We proceed to describe a variety of challenges in location-enabled m-services that relate to data representation, indexing, and on-line analytical processing style querying.

These are but a few of the many challenges posed by the developments outlined above. Wolfson et al. [17] have previously offered their complementary perspective.

2. Data Representation

To be able to effectively and efficiently offer new location-enabled m-services with high frequency, it is important to avoid software development strategies and software architectures where a new, monolithic, stove-pipe-like system is developed for each new service. With such systems, there is little reuse when a new service is developed.

Integrated Representations To obtain reuse of data across m-services, an integrated representation, or data model, of all relevant geo-referenced content must be developed. Such a data model will promote reuse of content and lower-level services when new services are developed. This may lead to more rapid development of services as well as to higher-quality services, both of which are important in order for a provider of location-enabled m-services to maintain a competitive advantage. For example, certain types of services, such as games, have a relatively short life time; new services must be made available quite frequently to retain the customer base.

To better appreciate the complexity of this challenge, it is instructive to lean that The Danish Road Directorate maintains more than 1000 attribute values for each position on each road in their database, which covers all of Denmark. A substantial fraction of these need to be reflected in an integrated data model.

These attributes are quite specialized in that they are closely related to the roads themselves. In addition, the data model must capture the "real content," which is much more voluminous and open-ended. For example, such content includes information about stores, e.g., their opening hours, available inventory, and current sales, and about cultural events, e.g., the artists, attendance prices, and seat availability.

While most geo-referenced content is stationary or changes location only at discrete times, some content changes continuously. The locations of the service users is a prominent example of the latter. Such content must also be captured in the data model.

The envisioned data model must be both conceptually simple and must also permit the provision of efficient services. It represents a substantial challenge to develop such a data model.

Euclidean Spaces, Obstacles, and Transportation Networks Much initial work on query processing in relation to moving objects has assumed that the geo-referenced objects that queries concern live in a 1-, 2-, or 3-dimensional Euclidean space (e.g., [1, 2, 4, 8, 11, 12, 14, 16]). Indeed, development of query processing techniques based on this assumption may lead to results that are very useful for some application contexts.

In other contexts, different assumptions may be more appropriate. Specifically, it is sometimes appropriate to expect some objects to constrain the locations of other objects [10]. For example, the movement of a ship is constrained by shallow water and land. Note that the constraining object's spatial extent varies across time, e.g., due to the tide and seasonal phenomena. As another example, the movements of joggers in a forest are constrained by fenced, private properties, swamps, etc. Little research in query processing has been reported that takes into account such constraints on the movements and locations of objects.

These blocking objects do not reduce the dimensionality of the space in which the objects are embedded [20]. In other application contexts, it is appropriate to constrain the locations of objects to a transportation network embedded in, typically, 2-dimensional Euclidean space [7]. This in some sense reduces the dimensionality of the space available for movement—the term 1.5-dimensional space has been used. Examples abound. Cars typically move in transportation networks, and the destinations such as private residences, hotels, and shops may be given locations in transportation networks.

While some proposals for query processing techniques have taken into account the presence of transportation networks, it is our contention that many more opportunities exist for developing query processing techniques that exploit the presence of such networks.

We believe that, stated generally, better performing techniques may be developed when it is possible to constrain the movement of objects to a network, than when objects move more freely.

It should also be noted that Euclidean distances are often not of interest in this setting. Rather, techniques that apply to settings with obstacles or network constrained objects will have to contend with other notions of distance. For some such notions, the distance between two stationary objects varies over time.

The kinds of queries one may expect also depend on the network more generally. Folklore has it that 80-90% of all automobile drivers move towards a destination. This suggests that drivers typically follow network paths that are known ahead of time. This statement may also lead one to assume that drivers have a particular interest in different properties pertaining to the path from their current location to their destination. They may want to know the remaining distance and the anticipated remaining travel time. In this context, typical queries may not be simple range or nearest neighbor queries.

Uncertainty Uncertainty is inherent to all or most geo-references of content and must be taken into account both in the representation of content and when querying the content. Uncertainty is perhaps most prominent in relation to the locations of moving objects [9, 18].

Object locations are sampled according to some specific protocol [19]. As examples, the users may disclose their locations when they desire a service, they may supply their locations at regular intervals, or they may keep track of where the service thinks they are and then issue location samples to the service exactly when necessary in order for the service's record to be within a certain threshold of the actual location. In this latter example, the threshold is dependent on the specific service desired. Because location data are obtained via sampling, complete traces of the objects' movements are unavailable; rather, the service only knows the locations of the objects at discrete times.

Additionally, the samples themselves are imprecise. The sample imprecision is dependent on the technology used and the circumstances under which a specific technology is used. For example, the cellular infrastructure itself, the positioning technologies offered by companies such as SnapTrack and Cambridge Positioning Technologies, and GPS and server-assisted GPS offer quite different precisions. And, for example, GPS technology is dependent on lines of sight to several satellites, which affects the robustness of the technology. In other words, the accuracy of the positioning is highly dependent on the user's location.

If a precise (but incorrect) trace is maintained for each user, queries may return suboptimal results. On the other hand, if an overly imprecise record of the positions is kept, query results will also be suboptimal. Maintaining a very accurate record of each user's trace will yield the best query results, but may also lead to poor query performance. The envisioned data model for geo-referenced content must maintain a representation that is adequately precise and that does not adversely affect query performance.

Further, it must be able to maintain content with different precisions. For example, some locations are obtained using GPS and can thus be obtained at at a precision of ca. 10 meters. However, for the user without a GPS receiver, the location can only be obtained using the cellular network, where the most precise location is approximately given by a cell, i.e., a precision of several kilometers.

3. Indexing

Indexing denotes a class of fundamental data management techniques that make it possible for multiple users to concurrently and efficiently retrieve desired data in very large databases. Indexing techniques are becoming increasingly important because the improvement in the rate of transfer of data between disk and main memory cannot keep pace with the growth in storage capacities and processor speeds.

Hyper-Dynamic Data In location-enabled m-services that involve large amounts of content, that rely on access to up-to-date information, and where continuous variables (e.g., the locations of users) are monitored, updates occur very frequently.

Due to the volumes of data, the data must be assumed to be disk resident; and to obtain adequate query performance, some form of indexing must be employed.

Traditional indexing techniques do not work well for hyper-dynamic data. Stated briefly, the indices have to be explicitly updated when changes occur to the data. For large, continuously changing datasets, e.g., those capturing the positions of moving objects, the constant updating of the indices would require very large computing resources, rendering the use of indices either impractical or totally impossible. Alternatively, the large volumes of updates and the mechanisms that regulate the concurrent use of the indices would combine to block the querying of these structures, also rendering them useless.

The fact that existing solutions can accommodate only relatively few updates presents a serious problem for the kind of services considered here (as well as for other services that rely on the monitoring of continuous variables via some forms of sensors).

Accommodating Continuous Change The challenge of how to obtain the normal benefits of indexing when the data being indexed change continuously is an important one. Two types of techniques may be envisioned: those that reduce the numbers of updates and those that make the processing of updates more efficient. We consider each type in turn.

One approach is to model the positions of objects as functions of time. This alleviates the need for frequent updates, as updates are only needed when the changes in function parameters and the time time since the last update combine to yield a difference between the real location of the object and the position believed by the database that exceeds the threshold dictated by the services being supported.

Several researchers have explored the use of linear functions for indexing the current and future positions of moving objects in 1-dimensional to 3-dimensional spaces [2, 7, 12, 14, 15]. The use of more general functions has yet to be explored, as does the combined indexing of the past, current, and future positions of moving objects.

Next, if the locations of moving objects are constrained to a transportation network, the network may also be exploited in various ways to reduce the number of updates. Many open issues remain.

Complementary techniques of accommodating frequent updates should also be studied. For example, buffer techniques may be applied [3]. Briefly, these remedy the inefficiencies of transferring blocks with little data between main memory and disk by buffering updates. The goal of using advanced buffering techniques is to guarantee that the I/O operations needed to maintain an index under updates completely use all the space available in disk pages for useful data. This goal should be reached without adverse effects on query performance.

As other examples, distributed techniques and large main-memory resident structures may be used aggressively to avoid the I/O bottleneck.

Another promising direction is to exploit approximation techniques to enable monotonically improving, as-good-as-possible answers to queries within specified soft or hard deadlines. These may enable querying of almost up-to-date data, which in turn may reduce the need for prompt updates.

4. Advanced Querying—Data Modeling and Precomputation

The kinds of queries considered in the context of moving-object indexing are mainly generalizations of the typical spatial database queries: range queries, nearest neighbor queries, incremental k nearest neighbor queries, mono- and bichromatic reverse nearest neighbor queries, etc. One generalization is to add time ranges to such queries. Another is to attach the queries to moving objects. Yet another is to make the queries persistent for some time range, meaning that the result is kept up to date for this time range, thus ensuring that the result reflects the movement of objects and changes caused by updates. Indexing structures as described in the previous section aim to meet the challenges of supporting such queries.

In the area of data warehousing and on-line analytical processing, focus is on entirely different types of queries, namely queries that involve the aggregation of very large amounts of data. In this context, precomputation is essential. For example, the total use of a service by county (the location from where the use took place) and month (when the use took place) could be precomputed and stored. Reusing this result enables fast answers to queries that ask for the number of uses of the service, e.g., by month alone, by county alone, or by quarter and county in combination. This is because the answers may be derived from the precomputed result alone; access to the bulks of data in the data warehouse is not needed.

On-line analytical processing in general and precomputation in particular have traditionally assumed static and relatively simple business data. This leads to two substantial, multi-faceted challenges.

In the multidimensional data models used for OLAP, *facts* represent the things that should be analyzed. Facts are characterized by *dimensions* that provide context and have associated *measures* that capture the numerical properties that should be studied. Existing models only support relatively simple data structures and are unable to fully capture the complexity of advanced geographical data types such as geographical regions, travel plans, transportation networks, etc. How to accommodate such data types in a multidimensional data model represents an interesting, challenging, and important challenge. Aspects such as imprecision and varying precision of geographical references also cause problems and should be supported.

The precomputation techniques that are being employed in on-line analytical processing were originally invented for environments where data processing was separated into distinct data loading and querying phases. In the applications we consider, bulkloading is inappropriate; rather, we are faced with rapid traditional updates.

This means that precomputation techniques are faced with some of the same challenges as are indexing techniques. Because precomputation is essential in ensuring interactive response times to advanced queries, meeting these challenges is of high importance if on-line analytical processing is to be used in conjunction with location-enabled m-services. We conjecture that some of the same general types of techniques as mentioned for indexing may be applicable here.

5. Summary

With the continued proliferation of wireless networks, based on evolving standards such as WAP and Bluetooth, visionaries predict that the Internet will soon extend to many billions of wireless devices, or objects. A substantial fraction of these will offer their changing positions to location-enabled services, they either use or support. As a result, software technologies that enable the management of the positions of objects capable of continuous movement are in increasingly high demand.

This paper aimed to identify selected areas of query processing where this development poses interesting research challenges. First, focus was given to the representation of location-referenced content, where the challenges concerned the integrated representation of large bulks of content with complex structure and inherently uncertain geographical references. Next, challenges related to the indexing of continuously changing phenomena were considered. Here, the central challenge was to contend with potentially huge amounts of updates. Finally, two challenges related to the use of on-line analytical processing in the context of the location-enabled services considered here were described. One was to model complex geographical data types in the relatively rigid multidimensional data models used for on-line analytical processing. The other was that of extending precomputation techniques to work together with very frequent updates.

Acknowledgments

The author thanks Augustas Kligys, Torben Bach Pedersen, Dieter Pfoser, Mindaugas Pleanis, Simonas Šaltenis, and Igor Timko for their collaboration on research related to this paper's topic.

The author's research in location-enabled m-services is supported in part by the Danish Technical Research Council, the Nordic Academy of Advanced Study, and the Nykredit Corporation.

References

[1] P. K. Agarwal, L. Arge, and J. Erickson. Indexing Moving Points. *Proc. of the PODS Conf.*, pp. 175–186 (2000).

[2] P. K. Agarwal and S. Har-Peled Maintaining Approximate Extent Measures of Moving Points. *Proc. of the ACM-SIAM Symposium on Discrete Algorithms*, pp. 148–157 (2001).

[3] L. Arge. External-Memory Algorithms with Applications in GIS. In *Algorithmic Foundations of Geographic Information Systems*, LNCS 1340, pp. 213–254 (1997).

[4] J. Basch, L. Guibas, and J. Hershberger. Data Structures for Mobile Data. *Proc. of the 8th ACM-SIAM Symposium on Discrete Algorithms*, pp. 747–756 (1997).

[5] M. Cai, and P. Z. Revesz. Parametric R-Tree: An Index Structure for Moving Objects. *Proc. of the COMAD Conf.*, (2000).

[6] J. Elliott. Text Messages Turn Towns into Giant Computer Game. *Sunday Times*, April 29, 2001.

[7] G. Kollios, D. Gunopulos, and V. J. Tsotras. On Indexing Mobile Objects. *Proc. of the PODS Conf.*, pp. 261–272 (1999).

[8] G. Kollios et al. Indexing Animated Objects Using Spatiotemporal Access Methods. TimeCenter Tech. Rep. TR-54 (2001).

[9] D. Pfoser and C. S. Jensen. Capturing the Uncertainty of Moving-Object Representations. In *the Proc. of the SSDBM Conf.*, pp. 111–132 (1999).

[10] D. Pfoser and C. S. Jensen. Querying the Trajectories of On-Line Mobile Objects. In *Proc. of the Second ACM International Workshop on Data Engineering for Wireless and Mobile Access*, pp. 66–73 (2001).

[11] D. Pfoser, Y. Theodoridis, and C. S. Jensen. Novel Approaches in Query Processing for Moving Object Trajectories. *Proc. of the VLDB Conf.*, pp. 395–406 (2000).

[12] C. M. Procopiuc, P. K. Agarwal, and S. Har-Peled. STAR-Tree: An Efficient Self-Adjusting Index for Moving Objects. Manuscript (2001).

[13] Ch. E. Ramirez. Games Find Home in Mobile Phones. *The Detroit News*, March 26, 2001.

[14] S. Šaltenis, C. S. Jensen, S. T. Leutenegger, and M. A. Lopez. Indexing the Positions of Continuously Moving Objects. *Proc. of the ACM SIGMOD Conf.*, pp. 331–342 (2000).

[15] S. Šaltenis and C. S. Jensen. Indexing of Moving Objects for Location-Based Services. *Proc. of the IEEE International Conf. on Data Engineering*, to appear (2002).

[16] J. Tayeb, Ö. Ulusoy, and O. Wolfson. A Quadtree Based Dynamic Attribute Indexing Method. *The Computer Journal*, 41(3): 185–200 (1998).

[17] O. Wolfson, B. Xu, S. Chamberlain, and L. Jiang. Moving Objects Databases: Issues and Solutions. *Proc. of the SSDBM Conf.*, pp. 111–122 (1998).

[18] O. Wolfson, S. Chamberlain, S. Dao, L. Jiang, and G. Mendez. Cost and Imprecision in Modeling the Position of Moving Objects. In *Proc. of the 14th International Conf. on Data Engineering*, pp. 588–596 (1998).

[19] O. Wolfson, A. P. Sistla, S. Chamberlain, and Y. Yesha. Updating and Querying Databases that Track Mobile Units. *Distributed and Parallel Databases* 7(3): 257–387 (1999).

[20] O. Wolfson, S. Chamberlain, S. Dao, L. Jiang, and G. Mendez. Cost and Imprecision in Modeling the Position of Moving Objects. In *Proc. of the 16th International Conf. on Data Engineering*, pp. 687–688 (2000).