Abstract

The location of the mobile user is fundamental to all location-based services. To enable these services the mobile middleware platform needs to access the user's location. Today a number of vendors like Ericsson and Nokia can provide mobile positioning services (MPS) that can be used to locate the user's mobile device. Push services, triggered by the user's location, also put heavy demands on the MPS.

GPS technology and network based positioning has been researched in detail. Great strides are being made in improving the performance and accuracy of determining the position of a mobile device. This paper looks at performance and scalability issues in accessing the positioning information from the middleware platform. We discuss an implementation where we allow the application to specify the accuracy of positioning required and then use the user's current location and velocity to determine the location caching policy.

1. Introduction

Location based services (LBS) are moving rapidly from the realm of research [1,2,3,4] to the main stream [5]. Industry analysis indicates that messaging is the most important feature in the current mobile applications marketplace but LBS will be the driving forces behind in the next generation of commercial services. A fundamental requirement for LBS is the availability of the location information of the user. Wide scale deployment of position determining equipment (PDE) in the mobile network is taking place [6,7] and we are also seeing the emergence of phones with built-in GPS receivers [8,9]. In this paper we propose that we need to cache the position information of the user in the mobile middle-tier platform and suggest policies to manage this cache.

In Section 2 we discuss the need for caching the location information in the middle tier. In the following section we introduce the concept of the application specified positioning accuracy requirement and how that is used in refreshing the cache entries. Then we briefly discuss location based events and the demand they put on location caching. We propose a caching policy that scales well for location based applications.

2. Location Cache

Why do we need a location cache in the middle-tier? The position information of the user’s mobile device will be made available by the carrier by deploying Mobile Positioning Services (MPS) on the internet. It will be protected using an appropriate privacy framework. Each access to the MPS to acquire the user’s position is expensive and the typical mobile application usage pattern is bursty, i.e. the user will periodically perform several operation in quick succession. Most applications also do not need the ‘exact’ position of the user. For example, when the user is looking for a gas station in the vicinity, it is perfectly acceptable to be inaccurate by several hundred meters. These two factors - the bursty access pattern and the lack of accuracy requirement suggests that a cache of the user’s position in the middle-tier would be extremely effective in not only improving the application response time but will be essential for the scalability of the MPS.

3. Cache Management

The location cache is quite different from file system or database caches. Some version of the ‘least recently used’ (LRU) cache replacement policy has been found to be extremely effective in data caches. In the cache of location information the entry (the position of a user’s mobile device) is valid only for a short period of time after which it is ‘stale’ and can be replaced. How quickly the position information becomes ‘stale’ is determined by how fast the user is moving. In the extreme case where the user is not moving the position information never becomes ‘stale’! The user may statically specify a transportation mode - walking, city driving, highway driving, etc. or the system could dynamically determine the velocity of the user. The situation is further complicated by the fact that different applications have different accuracy demands. A navigation application
might regard the user’s position information to be ‘stale’ if it is 10 seconds old whereas a business finder application will be perfectly happy with position information that is several minutes old. We therefore introduced the concept that the application has to specify a ‘Positioning Quality of Service’ (PQoS) when it is deployed. The PQoS specifies the accuracy of the user’s position information that is assumed in the business logic of the application. We also allow the developer of the application to override the preset PQoS for the application by specifying the PQoS value in each location acquisition request from the MPS.

When the position of a user is acquired from the MPS and entered into the cache, we also store the current time and the velocity. Some MPS provide the velocity value. If the MPS does not provide the velocity then we initially use a value based on the default transportation mode specified by the user. Subsequently, the average velocity is calculated based on the last two position and time values acquired for the same user. When a request for a user’s position comes from an application and we find the user’s position in the cache then we check if the entry is ‘stale’ based on the application’s PQoS value and the cache entry time and velocity. If the cache entry is ‘stale’ then a new location for the user is acquired from the MPS and the entry updated in the cache.

4. Location-based Events

Once we move beyond the first generation of simple location based services (e.g. find me a restaurant within 2 miles), ‘location based events’ (LBE) will be an important application area. An LBE is defined to be an ‘event’ (application or business logic) that is triggered by the change of position of an user or a set of users. For example, you are waiting for an important delivery and you would like to receive an SMS alert when the delivery truck is 15 minutes away from it’s destination. These types of applications put a significant load on the MPS because the location of the truck has to be accessed frequently to determine if it is within the 15 minute driving time region and your package is the next delivery. For these types of applications we have to consider another factor that we did not introduce in the simple position caching case discussed in the last section. Now we not only have to track the truck’s current location and velocity but also its distance from the destination ‘region’ specified in the LBE. If the truck is 500 miles away from the destination and moving at an average speed of 50 miles per hour then it makes no sense to position the truck frequently. Once the truck is within 50 miles of the destination the LBE application needs to know the position the truck more accurately. We have developed policies for position acquisition based on the current location, velocity and distance from the event ‘region’ for these types of applications.

5. Summary

The location cache is fundamentally different from the data cache. In a location cache a frequently used cache entry also needs to be refreshed after a certain period of time. This refresh period is determined by the velocity of the user and also by the distance accuracy requirement of the application. In this paper we propose an application driven cache refreshing policy that is geared specifically towards managing location information.

6. References