



# Automated Location Based Services – an Apisphere Whitepaper

Vytas SunSpiral, CTO  
Apisphere Inc.  
February 6, 2009

## Introduction

As location-based services for mobile devices proliferate, it is worth distinguishing between two general classes of applications: on-demand location based services (LBS) and automated location based services.

On-demand LBS, such as Google Latitude, are generally user initiated and involve requesting information or services relevant to the users current location. This form of application has seen the greatest number of new offerings because it is easy to implement and does not require an in-depth knowledge of location technologies or location accuracy.

Automated LBS focus on continuous monitoring of location and the automatic initiation of processes, such as message delivery, at the contextually right time and place. This approach is technologically more challenging to implement, but offers a wealth of innovative new applications that tend to be well suited for enterprise and retail needs.

This paper will explore the differences between these two classes of applications and then will focus on automated LBS, examining what makes them challenging, especially when utilizing mobile devices such as cell phones. The key components of a platform designed to support automated LBS will then be discussed, with a focus on supporting a wide range of current and future location technologies, networks, applications, and integration with existing enterprise systems and business models.

## On-Demand Location Based Services

The recent explosion of new location-based applications for GPS enabled smartphones has largely focused on the easiest applications to develop: on-demand LBS. These applications generally employ simple calls to the device's location API and then provide information or services based on the result.

Examples include providing local maps and locations of nearby services such as coffee shops, gas stations, movie theaters, local weather and traffic or in the case of Google Latitude - find your friends. These sorts of on-demand LBS have been easy to implement because they are robust to the noise and error that is inherent in any sensor data.

While GPS can be very accurate most of the time, occasionally it can also be wrong by large distances, from blocks to hundreds of miles. Errors and noise are a fact of life for all methods of location sensing, and will be discussed in more depth later in the paper.

### The Human Data Filter

On-demand LBS can largely ignore these errors because the user themselves acts as a data filter. This happens in a couple ways. If the location of the user is inaccurate, but the service that is provided is a local map with nearby destinations highlighted, as long as the map is local enough to where the user is they will localize themselves on the map and will still find the information useful.

If the returned location is so poor that it is not helpful to the user, they will notice this, because they can look at local street signs or other "ground truth" information. A user will tend to actively interact with the system at these moments, moving to a new location, perhaps with a better view of the sky, and requesting a new location fix. Thus the user, in both cases, acts as a very sophisticated data filter, rejecting bad data and working with the system to get the best data possible.

This involvement of the user in the filtering and interpretation

of the data is what makes the on-demand LBS robust to errors and noise in the location technologies, but is also the limitation of the approach. Fundamentally, only fairly unsophisticated applications can be built in this manner, and most of them have not found a good method to drive revenue other than the vague promise of including ads or as features of a phone that drives sales of the phone itself.

## Automated Location-Based Applications

The bulk of applications that are of interest to enterprises and commercial establishments, with real revenue potential and the ability to create new business models, require continuous localization and automation.

A first example would be an extension to the on-demand services listed above. Currently, a manager at Starbucks has no idea if someone has walked in the door because they found it by requesting the location of nearby coffee shops on their phone, or if they simply walked around and entered the first coffee shop they saw. An automated LBS platform would be able to integrate the user's request for near-by coffee shops, deliver them ads or coupons for specific stores, and then register which store they physically entered, creating a meaningful feedback loop on the couponing and advertising services which were deployed.

To do this, the system would need to automatically and accurately know which store the user entered, without the benefit of the human filtering the noisy data from the system. This is one of the first issues to make this class of applications technically more challenging.

When those technical challenges are properly handled, the range of possible applications that are enabled is exciting. From a user perspective, automated LBS are like having a location aware concierge monitoring their motion through the world and performing actions on their behalf at the appropriate time and place. Imagine the executive arriving at an important business meeting and having information, such as current

market data and headlines about all the parties involved in the meeting, dynamically gathered and automatically delivered to their phone.

The processes invoked by the automated LBS platform may not even directly involve the user or their device. Imagine a sales person leaving one meeting and the platform sending a courtesy email to their contacts at the next meeting informing them that they are enroute and providing an estimated arrival time based on current traffic conditions — with a little extra time for padding. In the processes of gathering the traffic data, alerts and detours could even be automatically sent back to the sales person if there was a known traffic problem on their route. These, and other applications, have already been built using The Apisphere GEM Platform<sup>1</sup>.

### Every Industry Touched by Automated Location Based Services

Over the coming years, almost every industry will find new uses for automated LBS, and will be changed by the technology. Staff and fleet management, with asset tracking and automatic timesheeting, is an obvious early adopter of these technologies. Retail, advertising, and customer loyalty programs will also find many new ways to engage with their customers in a location and context appropriate manner.

New tools for group and personal social interaction, communication, marketing, and safety based on location and presence are starting to be explored. Even beyond business and pleasure, location-based automated services will contribute life-changing benefit to the sick and disabled. For example, simply alerting caretakers of unusual movement by an Alzheimer's patient can save lives. Location-based automation will also be a key component in a network of assistive technologies that enable an ageing population to maintain independent lifestyles even as their physical capabilities wane.

They key to these applications is the ability for location to

be continuously monitored and reasoned about. This is the technological basis that enables the automated services to run in the background of a users life, performing the contextually optimal actions when and where they are most appropriate.

### Continuous Localization: In the Car, and on the Phone

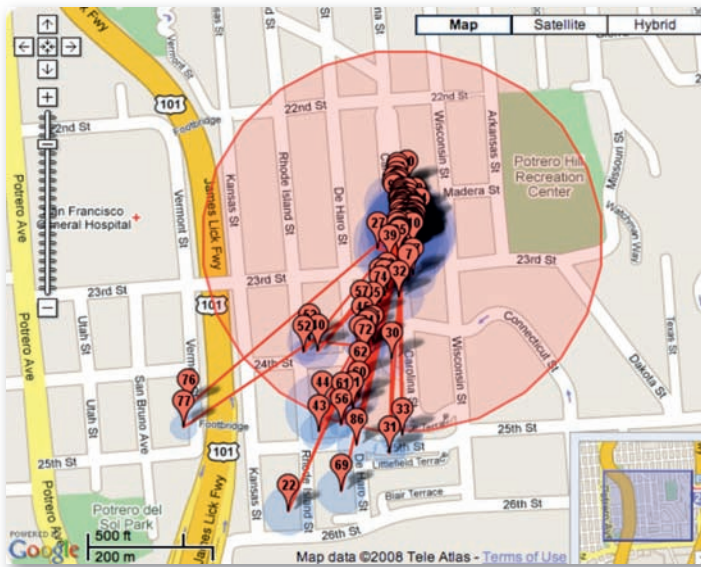
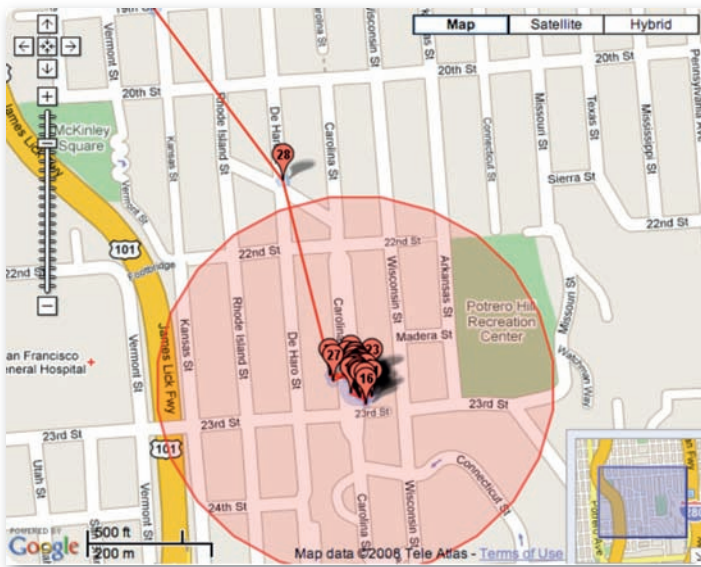
At the heart of all these applications is the ability to perform accurate continuous localization. One of the first widely adopted continuous localization systems has been the in-car navigation assistant.

These systems do an amazing job of localizing cars on the road to within meters while giving precise directions, and have rapidly become integral to our lives. Thus, it comes as a great surprise to many that this same level of accuracy is not easily available on a GPS enabled cell phone. The challenge in implementing continuous localization on cell phones stems from two sources: differences in device capabilities and difference in how and where the devices are used.

### Quality of Location Data

Location quality is affected by a significant difference in device capability between in-car navigation systems and cell phones. Cars have plenty of power and tolerance for weight and size. Cell phones are designed to be small and compact, and must keep tight control over power usage in order to maximize battery life. As a result, cell phones use smaller antennas which run at lower power levels, leaving them weaker signal to noise ratios.

Even more significantly, they tend to gather data at lower rates, again to save power. Yet, since data is noisy, a common way to deal with that noise is to gather data at a high rate so that it can be filtered and smoothed. A low rate of data collection will cause such filters to be less responsive. Dealing with these issues is possible, but requires an intimate knowledge of how GPS works in order to implement the statistical modeling algorithms required to make best use of the data that is available.



**Figure 1.** Two GPS based location tracks captured by the same BlackBerry Curve at different times. The red circle has a radius of 1/5 mile. In the top track, the user has left by car and the data is valid. In the bottom track the phone is sitting on the kitchen counter the whole time and the location API is returning poor quality data. Note that the blue circles indicate the accuracy of the data as estimated by the location API.

Another factor that makes continuous localization easy to perform for an in-car navigation system is that the car is almost always outside with a clear view of the sky, and stays on roads, which are well mapped. Cell phones, on the other hand, can travel anywhere – into buildings, on the sidewalk, into parks, on roads, etc. This affects location data in two ways.

First, GPS works best when the device has a clear view of the sky and can see many satellites. Being indoors degrades the location information and makes errors more likely, if a signal is available at all. Likewise, a cell phone in a backpack or purse is more likely to report noisy erroneous location data. The second advantage that in-car navigation systems gain from their limited mobility is the assumption that the car is on a road network. This assumption can be used to further reduce the error and noise inherent in GPS data by using statistical techniques to collapse the area of location uncertainty onto the road network and predict near future motion.<sup>2,3,4</sup> This assumption is not available to cell phones, since humans are not constrained to known road networks and their motion is less predictable.

### On-Device Location API's Insufficient

The differences between in-car navigation and cell phones make the task of continuously localizing cell phones an unexpected challenging task for most developers. Figure 1 illustrates this by showing two different location tracks captured by a system that naively calls the location API on a regular interval and plots the returned data. What further complicates this picture is that, despite common API's, the behavior and quality of data varies across handsets depending on the underlying GPS hardware (or other sensor) and how the specific API implementation processes the raw sensor data. Thus, any application developer who wishes to provide automated location-based services across a wide range of devices must understand the behavior of each of those devices.

In the top track of Figure 1, the user has left their house by car and we can see that the system has done a fine job of locating them, both in the house and while driving. In the second track,

taken some hours earlier, the GPS satellites were in a different configuration and multipath signal errors were dominating the data. During the entire period that this second data-set was collected, the phone was sitting stationary on the kitchen counter. Even though the GPS calls were returning locations with estimated accuracies on the order of 30-50 meters (blue circles), they were incorrect by up to 1/3 of a mile. Clearly the naïve approach of extending the on-demand LBS model by regularly calling the on-board location API will not make a good tracking system by itself.

### Robust Continuous Localization

There are three aspects to creating a robust continuous localization system designed for cell phones and other mobile devices:

1. Deep knowledge of the location technology.
2. A change in how location information is conceptualized.
3. Fusion of multiple data sources.

Abstract location APIs (such as JSR179) are compelling to the developer of simple on-demand LBS who does not have, or want to have, an in-depth knowledge of GPS or other location technologies. In contrast, construction of a system for continuous localization requires intimate knowledge of the underlying localization technology. The noise and error in the system can be greatly reduced by understanding its sources and controlling the device in an optimal manner. Further, the noise can be modeled and appropriate statistical algorithms used to compensate for it. These algorithms must be tuned to the behavior of different underlying location technologies, which may even vary between models of the same brand phone, based on what GPS chip set is used.

The second step in developing a robust continuous localization system is to shift how location information is reasoned about. While it is compelling to request a location and get exact latitude and longitude values back, the fact is that those numbers have limited value in themselves. Any robust system must shift from treating location as a known fact to treating location as a probability distribution over an area. The location

APIs provided to application developers must be built up on these concepts and enable the developers to define actions and automations to occur when devices are in known zones with application definable confidence. These statistical approaches, many of which have been developed in autonomous robotic location and navigation systems,<sup>5,6</sup> isolate the application developer from the sensor noise and uncertainty that occurs if the raw location data is handled directly.

Sensor fusion is a powerful technique for dealing with the noise and uncertainty of continuous localization on cell phones. Many devices have multiple methods by which they can be located, both on-board and off. One approach is to use the best available modality, such as using WiFi triangulation indoors when GPS is not visible. More subtle techniques allow one to use information from one sensor to reject noise in another sensor by computing the likelihood of physical motion of the device given the change in the respective signal environments.

This sensor fusion concept can be extended to location information gathered from entirely different systems, such as indoor positioning systems, carrier-side location data, in-car navigation systems, etc. Ultimately, any form of location data can be fed into the system and selectively fused together to provide the most accurate and cost-effective estimate of location. This enables the location-based automation to work seamlessly across all the environments that we move through, and across the wide range of mobile devices that we use. Since all these devices and location technologies can be expected to mature and evolve rapidly over the coming years, this approach also allows flexibility in how applications evolve with the technology. Rather than tie an application to a single localization technology, it becomes the role of the automated LBS platform to integrate with the newest technologies. This isolates individual applications from the technological churn, yet enables them to take advantage of improved availability and data quality of future location technologies.

## Automated Location Based Services Platform

As the previous discussion illustrates, continuous localization is more complex than repeatedly calling an on-board location API at some fixed interval. Rather, a robust system for continuous localization will integrate data from a variety of sources. Likewise, automated LBS require more than just continuous localization.

A robust system for automated LBS needs to support:

- Many sources of incoming location data,
- Tools for defining location triggered events and associated actions,
- Support the execution of those actions,
- Manage the storage and analysis of data,
- Integrate with external data sources and processes,
- Handle privacy concerns,
- Provide data security,
- Manage costing and payment issues,
- Provide a stable environment that enables the applications to work across a wide range of constantly evolving mobile devices, location technologies, and regulatory environments.

Figure 2 illustrates how robust automated LBS can benefit from a central platform to manage the integration of the many components that support the individual applications.

### Integrated Location Technologies

The bottom row of figure 2 shows how the platform can integrate location information from a variety of sources and systems. At the simplest level this allows for the use of the most accurate system currently available for the particular user. One example would be the transition to the use of data from an in-car navigation unit, due to its superior data quality, when a user starts driving. Likewise, the platform can manage the hand-off between on-phone GPS data and an indoor localization system. Multiple sources can also be fused into a single high-resolution estimate of the users location that is

better than any single source. Finally, costing models can also be applied where pay-per-dip LBS are only used when other sources of location information are not available or drop below desired quality levels.

This fluid use of multiple sources of location data is critical to the deployment of automated LBS that work seamlessly though all the environments that a user may move and on all the devices that may be deployed. As it stands now, any application that wishes to interact with a broad user base needs to be ported to many different handsets, each with unique features and idiosyncrasies. A recent study showed how deeply fractured the handset market is, with 9 out of the 10 most popular handsets in the US accounting for 2% or less of the market each<sup>7</sup>. Having the automated services platform be responsible for integration with the many available devices frees the application developer from the constant struggle to keep up with new devices. Instead, applications written for the platform will work on new devices with minimal effort as those devices are integrated into the platforms eco-sphere.

### Integrated Core Services and Web Services

The middle layer of figure 2 shows how the platform can integrate with many other services beyond the location technologies. It is at this layer that all the supporting technologies can be integrated. Some of these services may be generic across most applications, such as data storage and analysis, billing and payment services, and specialized handling of privacy concerns to insure that only authorized applications gain access to users location data. By providing a core set of services which most applications will require, the platform further simplifies the process of rapidly building novel applications and services.

Furthermore, the platform may integrate any external service with either a public or contractually accessible API. Thus, an application that wishes to incorporate information such as current weather, or map data, or stock prices can do so easily through a web services interface.

### Location-based Applications



**Figure 2.** Examples of the many systems which must be integrated by the Automated Location Services Platform in order to support a wide range of location based applications and services.

Finally, the top layer of Figure 2 shows that many different types of applications can be supported by the platform. By providing a consistent API and development environment to all the underlying tools, application development can occur rapidly. While the automated LBS platform was designed with the intent of supporting advanced applications and the continuous localization they require, on-demand location applications and services can also benefit from integration with the platform. Almost all LBS have the need to manage server interactions, handle privacy concerns, handle billing and payment services, support processing and execution servers, and store and log data.

Furthermore, even the simplest on-demand application will benefit from having a stable development environment and API that makes integration with a wide variety of devices and

carriers easy.

Likewise, it should be pointed out that applications do not need to be new stand-alone applications. Existing enterprise systems may be extended to include location-based functionality by using the APIs provided by the platform. Examples of this are the recent integration of Salesforce.com and Microsoft Outlook with the Apisphere GEM platform.

#### User Adoption Advantages

Application developers who hope to reach a wide user-base also benefit from using a common platform because it will ease user adoption and privacy concerns. Imagine, for a moment, the user's experience if each new location-based service or application is implemented as a stand-alone system. The users will be bombarded with requests to install new



software, each of which will be a somewhat different process, and they will have to evaluate each new offer on its protection of their privacy concerns. Contrast this with a common platform enabling a simplified user experience, and new services that can be added with simpler opt-in agreements that extend the range of products and services they are already engaged with. Furthermore, the common platform can build recognition for trust-worthy handling of location and other sensitive personal data, providing the user with a standard set of tools to manage how their location information is distributed. Similar to the emergence of trusted payment services like Verisign enabling the mass adoption of on-line shopping, the mass adoption of location-based services will be enabled by user trust in the security of their location information.

### Location Based Services Eco-System

Just as the automated LBS platform enables the easy integration of many location technologies and support services, it also enables the evolution of a thriving eco-system of location-based services and applications.

There are two major forces that enable this eco-system to emerge. The first is that the common platform enables applications to build on top of each other, reusing existing technologies and enabling the easy integration of new services as they emerge.

Even if a company's current business plan calls for a simple stand alone location service, they may find that the rapid evolution of the underlying technology will cause those plans to change in the near future so that they can integrate with new offerings. The second major factor in supporting the eco-system is the ease by which new applications can be developed. Since most of the hard work and system maintenance is performed by the location automation platform, application developers are free to focus on their core business logic and area of expertise. Using the provided tools it becomes quick and easy to build and experiment with new ideas. This ease of experimentation and development

is the fertile ground required for innovation and the quick deployment of new ideas required for success as companies in every industry rush to explore how location-based services will revolutionize their business.

### References

- <sup>1</sup> Details available at <http://apisphere.com>
- <sup>2</sup> Taylor, G., Blewitt, G., Steup, D., Corbett, S., Car, A., "Road Reduction Filtering for GPS-GIS Navigation," *Transactions in GIS*, 2001, Vol 5(3), pp 193-207
- <sup>3</sup> Bernstein, D. and Kornhauser, A., "Map matching for personal navigation assistants," *77th Annual meeting, The Transport Research Board*, Jan 11-15, 1998, Washington, D.C.
- <sup>4</sup> Civilis, A.; Jensen, C.S.; Pakalnis, S., Techniques for efficient road-network-based tracking of moving objects," *Knowledge and Data Engineering, IEEE Transactions*, vol.17, no.5, pp. 698-712, May 2005
- <sup>5</sup> Fox, V.; Hightower, J.; Lin Liao; Schulz, D.; Borriello, G., Bayesian filtering for location estimation," *Pervasive Computing, IEEE*, vol.2, no.3, pp. 24-33, July-Sept. 2003
- <sup>6</sup> Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics* (2005, MIT Press.)
- <sup>7</sup> [http://www.nielsen.com/media/2008/pr\\_081212\\_download.pdf](http://www.nielsen.com/media/2008/pr_081212_download.pdf)

©2009 Apisphere, Inc.



**Apisphere, Inc.** 2107 Dwight Way, Suite 2100, Berkeley, CA 94704-1005 USA  
Phone. 510-868-2655 | Fax. 510-868-1741 | [www.apisphere.com](http://www.apisphere.com)