Title: The only game in town.
Authors: Eric Legge-Smith, Grant McKenzie, Matt Duckham
Affiliation: Department of Geomatics, University of Melbourne

Intro: The gaming market continues to hold huge potential for location-based services. But developing new location-based games is not as much fun as it might seem.

Body:

Location-based gaming (gaming in which play depends on tracking players' geographical movements) is growing. However, the technical challenges facing location-based game developers may still be inhibiting the breakthrough from niche activity to commercial success.

Event the simplest location-based games can be great fun. The need for players to move around the game space and explore their environment, at the same time as interacting with other players, makes such games immediately engaging and surprisingly addictive. One only has to look at the vast video games market, worth more than US$10bn, and the staggering recent growth of online games (World of Warcraft now boasts more than 9 million subscribers worldwide) to appreciate the possibilities for online, location-based games. Increased availability of low-cost PDAs with colour screens and high quality audio, operating systems that support downloading of games, and the growing coverage of broadband wireless networks in many metropolitan areas, all seem to provide an ideal technological environment for location-based games. So why are there no notable commercial location-based games available today?

The most successful and well-known location-based game is geocaching (an outdoor treasure hunting where players use GPS to caches of "treasure" hidden by other players). Geocaching does not require real-time game playing capabilities. Geocachers can download information about geocaches in advance of playing, and subsequently use a GPS to find the geocache. Consequently, part of the success of geocaching can be attributed to its technical simplicity: geocachers need only a (wired) Internet connection at home and an ordinary consumer GPS unit to play.

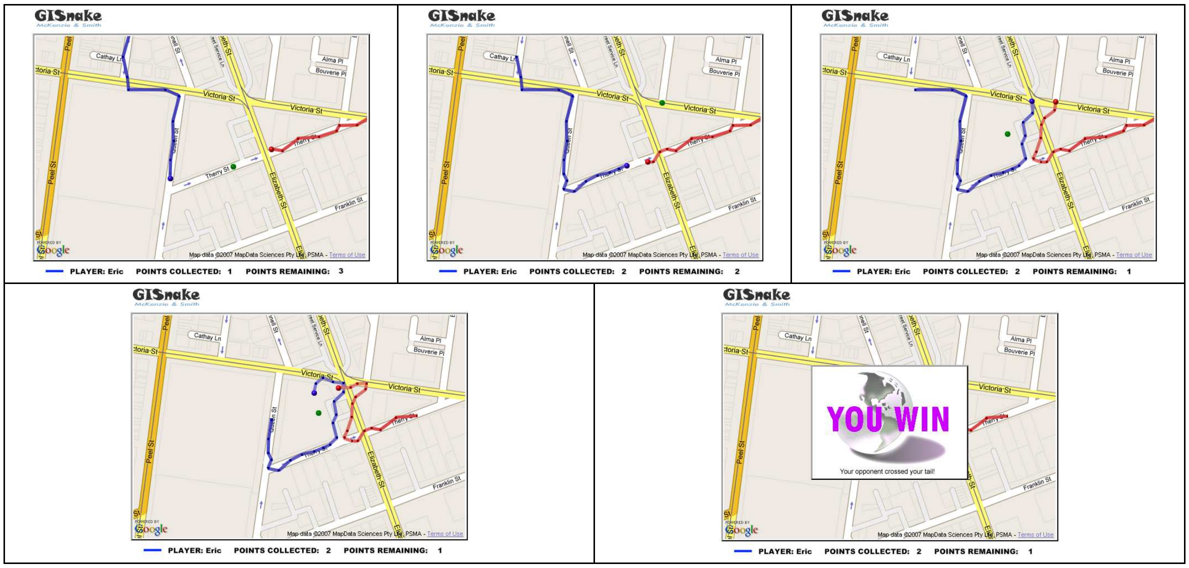
Credited as the first commercial location-based game, the massively multiplayer "BotFighters" achieved high levels of popularity in its country of origin, Sweden, as well as some other countries, like Russia and Finland. Botfighters is played on cell phones, and takes advantage of the cell-phone network as a simple positioning system. Players attempt to "kill" other players by "firing" SMS messages at each other. The network CGI (cell of global identity) is used to determine the approximate location of players, and whether they are close enough to get a good shot. While Botfighters is a near real-time game, its technical simplicity (players only need a GSM cell phone to play) has undoubtedly contributed to its success.

Building on these encouraging models, more recent developments in real time location-based gaming include Pac-Manhattan (http://www.pacmanhattan.com), Human Pacman (http://news.bbc.co.uk/2/hi/technology/4607449.stm) and ARQuake (http://www.wearables.unisa.edu.au/projects/ARQuake/www/index.html). These games aim to incorporate greater complexity, for example combining real-time gaming with GPS positioning and using sophisticated AR (augmented reality) interfaces. However, these more advanced games are still largely confined to university research labs. The task of translating them into a commercial environment appears to present substantial technical hurdles.

One of the biggest impediments to next generation, real time location-based games is the complexity of game development. "Gluing" together the spatial technologies required for location-based gaming, including GIS and positioning systems like GPS, with non-spatial technologies, such as wireless communication and mobile devices, presents developers with a major headache, even when only developing a prototype.

As an example, in the Geomatics Department at the University of Melbourne we recently developed a prototype location aware mobile game, called GISnake, based around the popular 1980s arcade game "Snake". In GISnake, the objective is for the player or players to move around a real geographic area grid collecting "apples". Collecting apples increases a player's score, as well as the length of a player's "tail". Players who cross another player's or their own tail lose the game, while the aim is to achieve the highest score possible.

In GISnake, the geographic play area is real. But the apples and the players' tails are virtual, and only displayed on mobile computers or PDAs carried by the players. Each player's visual displays are continually updated via a wireless communication network to reflect changes in their own and other players' tails and positions, as shown in the sequence of screenshots.



PLAYER: Eric POINTS COLLECTED: 1 POINTS REMAINING: 3

PLAYER: Eric POINTS COLLECTED: 2 POINTS REMAINING: 2

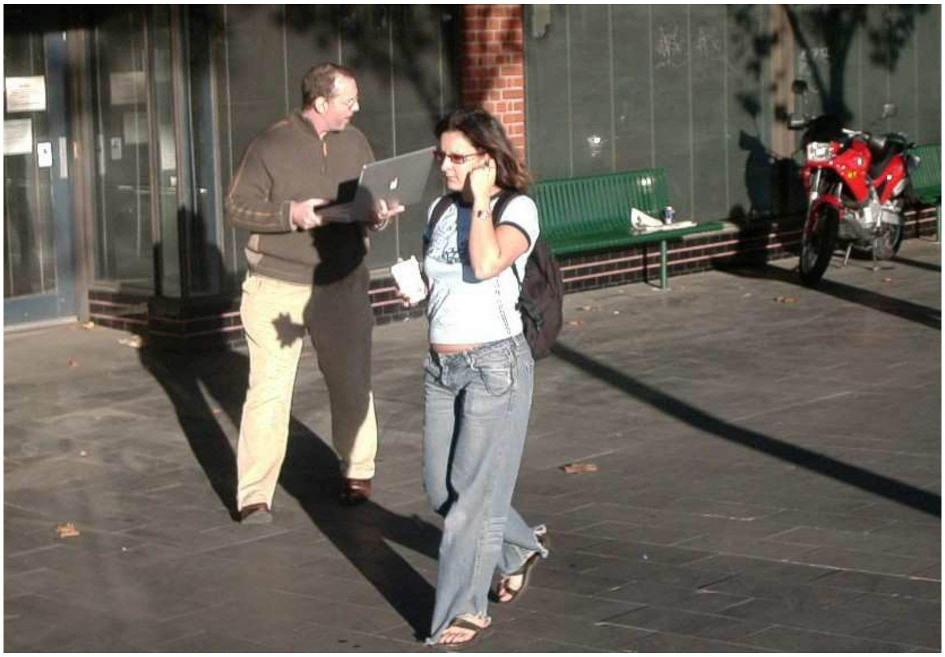
PLAYER: Eric POINTS COLLECTED: 2 POINTS REMAINING: 1

PLAYER: Eric POINTS COLLECTED: 2 POINTS REMAINING: 1

PLAYER: Eric POINTS COLLECTED: 2 POINTS REMAINING: 1

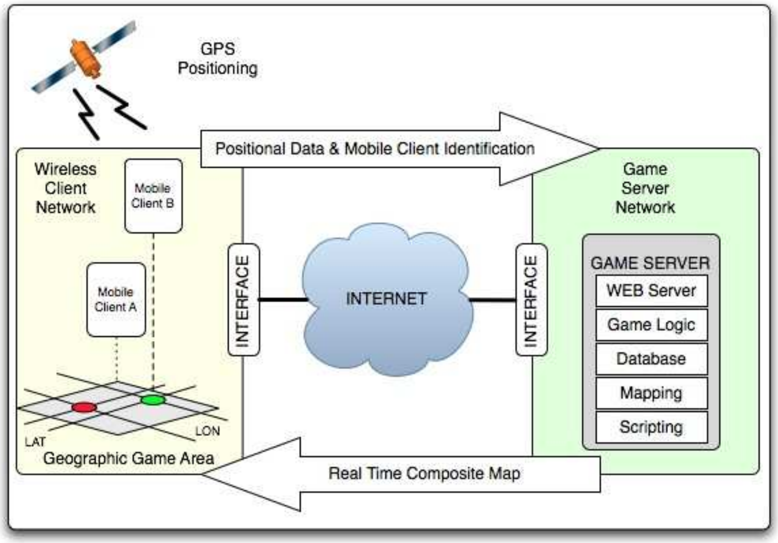
Sequence of screenshots for GISnake game interface

Even for our prototype, which used heavier laptops as mobile devices (pictured), this game works remarkably well and can be good clean fun. Racing around the city, dodging other pedestrians, finding the best route through the streets to collect all the apples, and devising strategies to box in your opponents with your lengthening tail, all make for excellent gameplay. Part of the appeal of the game is that it can be played anywhere: your local neighborhood, a busy city center, parkland or open spaces. The different characteristics and constraints to movement imposed by the different game environments lead to very gaming experiences.



Hunting for the next token in the prototype GISnake (using laptop in place of PDA)

However, the technical challenges facing even such a relatively simple prototype are formidable. The overall architecture is conventional enough (see architecture diagram). The geographic position of each player ("mobile client") is determined using GPS. Using customized software on the mobile client, each player's latest position is then relayed every few seconds back to the game server via the Internet (using either WiFi where available or packet switching cell-phone network). The game server integrates a number of components: scripting utilities (such as tools to parse NMEA strings and project coordinates); a map server, to produce maps of the player's immediate vicinity; a spatial database, for storing the player's current locations and their tails; a customized game engine that contains the game logic and rules of the game; and a web server, to communicate the latest game data in a form that can be displayed by any web-enabled mobile device.



Architecture of GISnake

An additional objective of developing GISnake was to use free and open source software for the different components as far as possible. The web server is Apache; the database mySQL; the scripting language PHP; and the map server was originally UMN MapServer (with Google Maps API later tested and used). The entire game server runs on a Linux PC.

Although each of the components in this architecture is, on its own, simple enough to understand and customize, the primary technical challenge of developing location-based games comes from integrating these components in such a way that very near real-time results can be returned to the user. Even relatively short delays dramatically affect the playability of the game, as users have to stop and wait to see if they have eaten the apple, bumped into an opponent’s tail, won or lost. For example, while there are many common map server solutions, only after a period of trial and error did we settle on one that was in practice fast enough for the game (using Google Maps API).

Some of the technical challenges arose from the diversity of mobile client devices (laptops, tablet PCs, PDAs, mobile phones) common today, which appreciably complicates client software development. Other challenges are explicitly spatial. The level of accuracy and precision of low-cost GPS, particular in urban gaming environments, means that unfiltered GPS data can lead to game errors. For example, unfiltered GPS errors in early versions of the GISnake prototype would mistake a stationary player as moving and even crossing his/her own tail. Consequently, making the game playable required the development of carefully tailored position filters that minimize such errors, while still being able to discern small enough player movements to make the game exciting.

The game logic for the game can also be surprisingly difficult to construct. Designing exciting multi-player games is a complex task. The process is especially challenging for location-based gaming, where the game itself is played in a geographic space. When developing GISnake, the development process included an intermediate stage of implementing a game simulation tool. Using this tool, the development team could test out aspects of GISnake at their desks by moving “virtual players” around a simulated game. However, in the later game development phases it is unavoidable that developers have to physically leave their desks and offices to test the game; a process which is fun, but time-consuming and labor intensive.

Despite these challenges, the experiences with GISnake showed us that it is possible for a small team to develop a sophisticated location-based game prototype. The levels of technical expertise currently required for this are very high. But, as new standards and development tools for LBS and location-based gaming emerge and take hold (such as Ericsson’s increasingly popular mobile positioning system, MPS, software development kit), it is to be expected that today’s technical problems will be tomorrow’s plug in components. Our experiences show that even relatively basic location-based games can be compelling and enjoyable. By helping to address some of the spatial aspects of the technology, spatial information science has a real chance to contribute to what may soon grow into a multi-billion dollar industry.