Cooperative mobile networks - Application to public transport

Vicente Acosta, Juan Pablo Revello, Ignacio Tisnés,
Leandro Scasso, Ricardo Rezzano, Ariel Sabiguero
Instituto de Computación, Facultad de Ingeniería
Universidad de la República, Uruguay
Email: {vicenteacosta3,revellojp,ignaciotg}@gmail.com,
{lscasso,rrezzano,asabigue}@fing.edu.uy

1 Introduction

ARTBus is a distributed and collaborative solution, that aims the problem of real time, best effort, transport fleet tracking without any support from transport companies or governments.

It relies on the sensors and the connectivity provided by smart devices owned by users of the public transport system.

The solution provides users with social tools to interact and create bonds with others sharing their path daily.

This project is based on a grade thesis work, coordinated with CUDEN (Collaborative Centric User-Device Networking) project, whose *"main objective is to provide support for ad-hoc cooperative communities between various devices to facilitate collaborative activities by extending users' physical experience and capabilities. (...) To this end, we will develop data processing and context modeling techniques to analyze users' social and surrounding environment information provided by various devices ranging from enhanced sensors to common products as smart-phones."*(STIC-Amsud.)

The Android operating system was chosen as the development platform for this project, given that it is open source, contains a powerful development API, has been one of the fastest growing platforms lately(StatCounter Global Stats.) and is the most used smart phone platform in South America.

Having chosen this platform the solution was developed using publicly available services such as Google Maps and Google Accounts. This project is work-in-progress at the time of the writing.

2 The solution

The solution's main function is to allow the user to know the position of the buses he is interested on using the information provided by other users. Optionally, more information could be provided regarding the status of said buses and the transportation system in general.

Despite having information available about the transportation system in our city, a decision was made not to use any of it for route gathering. This decision was made so that the same solution would be deployed in a different city and it would be able to model the local transport network relying solely on the information provided by the users to determine the available bus lines, bus stops, etc.

The next sections explain the architecture and the communication scheme used by the solution.

2.1 The architecture

As a device centric solution, developed as an Android application, it can be explained by the roles of the devices. One of the roles is mainly a querying role where the user can acquire the information known by the system about the state of the transportation network. This information is, in turn, provided by these same devices acting under the second role, where the devices publish information in the system about information known to them (for example, the bus a device is currently on and its position). Both roles can be active simultaneously in each device.

Another component of the solution is a server with the responsibilities of handling all information, managing the communications in the system and summarizing the data provided by the users.

In addition to this, the solution contains a social component which enables user to exchange messages among them serving both an informative function (letting other people know about a specific situation in the transport system), and an entertainment one.

The basic idea is that this solution offers relevant information about the transit system and entertainment to the user in exchange of CPU time, battery, mobile bandwidth, and sharing the information known to its device. Also, the social services are an extra reason for the users to give us their resources.

2.2 Communications

Users will select which information they are interested in (for example the position of all buses known belonging to a certain bus line). As many users will be expecting the same information, groups were defined according to that interest. The server will then send multicast-like messages to those groups according to their interests.

A custom protocol was made to allow this, which allows the server to send multicast-like messages to all users subscribed to events from a certain entity (for example, a bus line). This increases performance significantly because the processing is done as determined by the server, and then the result is delivered to all clients simultaneously. By using these multicast-like messages, a clean and efficient solution was accomplished.

At the moment the server executes two components which handle the data processing logic of the solution and the delivery of messages to subscribed clients, but they can be easily separated in different servers to improve the scalability of the solution.

All communications between the component responsible for data processing logic and the devices goes through the message delivery component. The latter manages the subscription to events defining topics dynamically in a way that is transparent to the end user. Thus, the message delivery component is responsible for ensuring that each client receives all and only the information requested by it.

3 Current status and roadmap

Currently, a third version of the solution has been completed, still in a prototype stage, including most of the core functionalities. One of the main objectives in this phase was to develop heuristics based on data provided by sensors present in Android devices to infer the status of the user with regard to the public transportation system. For example, these heuristics can detect when the user arrived at a bus stop, or when the user got on or off the bus.

Previous versions of this application were released to students in our University, both for testing and data retrieval purposes. During this experimental phase, 375 bus trips were recorded, providing more than 50.000 measures of points in the routes. The data collected was used as an input to design, calibrate and test the heuristics developed.

In previous versions, the application relied heavily on the user to input relevant events manually. For example, it was required for each user to tell when it is standing at a bus stop, gets on a bus, or gets off it. This cannot be completely avoided due to the self-learning property of the application, but the heuristics developed help automating some tasks when possible. Although progress has been made in this area, better heuristics can be developed to improve the range of cases for which automatic detection can be made.

The inclusion of the heuristics is important as it will enable the solution to function without requiring user interaction throughout the cycle. In an ideal scenario (which has not been implemented yet) the user would be able to use the application's information as needed, but no conscious decision would be required, when the user gets on each bus, to share the information known to them.

A master thesis is being developed also, whose objective is to take all the raw data gathered by this application and construct from it a model of the area's transportation system, overcoming various problems related to collected data quality. The final plan would be to use this information as an input for new heuristics.

Current prototype is still not attractive enough for spontaneous usage. After some trips, volunteers stopped using the application, showing us that, even well motivated test subjects found that the system was not ready yet for automated usage and data gathering. Social aspects were interesting, but, an insufficient number of users placed them in empty chat rooms. Despite the attractive of chatting with unknown people that share a bus line, more people is required in order to make it more interesting.

A similar problem is found when using the application to query the state of the buses belonging to a certain bus line. As few users continue to use the application, it is difficult to encounter a situation where the user performing a query receives the expected data. More ways of stimulating the users must be found to obtain their support.

References

STIC-Amsud. *Collaborative Centric User-Device Networking* URL `http://www-npa.lip6.fr/cuden/`.

StatCounter Global Stats. *Mobile OS (Operating System) Percentage Market Share Worldwide.* URL `http://www.areppim.com/analysis/stats/stats_mobiosxtime.htm`.

StatCounter Global Stats. *Mobile OS (Operating System) Percentage Market Share South America.* URL `http://stats.areppim.com/stats/stats_mobiosxtime_sam.htm`.