

DroidLab: mobile sensing made easy, fast and cheap

Balázs Lajtha, Rolland Vida
Department of Telecommunications and Media Informatics
Budapest University of Technology and Economics
Email: {lajtha.balazs, vida}@tmit.bme.hu

Abstract—We are living in a parallelized world. What was thought to be impossible for one can now easily be achieved by many. On one hand, we have the cloud: an army of obeying faceless machines. On the other hand we have the crowd: A colorful collection of volunteers. Clouds are built and sold but to control the crowd, each member of has to be won over.

Keywords—

I. INTRODUCTION

“Crowdsourcing is the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people in the form of an open call.” This was the initial definition of crowd sourcing when first used in 2006 by Jeff Howe and Mark Robinson. Two main orthogonal problem sets can be solved using crowd sourcing: data gathering and data processing. When a data gathering problem is distributed among members of the crowd we use the term crowd sensing or participatory sensing, while for distributed computational tasks we use the general term crowd sourcing.

Another way to categorize crowdsourced applications is whether the contributors profit directly of the gathered or processed information. Wikipedia uses the power of the crowd to provide a service that can directly be used by the contributors. When it comes to crowd sourcing platforms, creating a framework to acquire information is much easier than creating a generic framework that is able to present users with the results. In many cases the beneficiaries of the crowd sourced tasks are different than those providing the data. DroidLab targets these scenarios.

Probably the most popular crowd sourced computational problem was Seti@Home, a project lasting several years and aimed to ??? starts and stuff, extraterrestrial ???. This problem was divided into a large number of subtasks and distributed among the willing participants. Users could set a Seti screensaver that download and solved these subtasks when the user was away. Seti’s success came from it’s innovative design and from the interesting task that it performed. From the point of view of the volunteer, this was a low cost, low gain activity. PCs back then had long startup times and high standby power consumption, once Seti was installed it didn’t pose a visible overhead, and provided the user with insight of the performed work. Some early gamification elements were also implemented like leaderboards.

A somewhat different example is the Google Image classification game. In this game two players cooperated to solve a simple puzzle. An image was displayed in both player’s browsers. The first player had to find five words that described the image. Then the second player had to guess those words.

This human image processing game allowed Google to improve Image Search results. The players were motivated to play the game for the game itself, and for added gamification elements.

Both examples involved utilization of computational resources owned by the users to perform a highly parallelized task that the users perceived as common good, but didn’t benefit directly from the results of their work. In both cases, users received an input that they processed based on a given algorithm. The tasks were not personalized. Seti@Home’s tasks were solely performed by a machine, while Google Image games had to be solved by human users. In both examples users had extrinsic motivators.

Participatory sensing involves data collection outside of the system. Sensing can be automated or can involve the active participation of the user. Before smartphones, sensing was either done through expensive Internet-connected sensors or with the participation of the user through fixed personal computers. Sensors had to be deployed, and sensing involving the user had large delay and required considerable effort from the part of the user. The evolution of smartphones eased many new complex sensing applications. Smartphones have the connectivity that enable real time, high bandwidth communication. The devices can access precise location data, and are equipped with a wide range of sensors from cameras and microphones to accelerometers, gyroscopes and barometers. New mobile operating systems offer many seamless ways for the sensing system to prompt for user input.

To exploit these possibilities we offer DroidLab. DroidLab is a cloud backed mobile framework still under development, that aims to solve key challenges of mobile crowd sourcing. In this article we will present successful and promising mobile crowd sourcing application and crowd sourcing frameworks. We will identify the key challenges of mobile crowd sourcing and define the use cases for our framework. Then present the architecture and design of our solution and conclude with the evaluation of the current framework and define the future works.

II. RELATED WORK

We already mentioned two successful examples of crowd sourcing. Seti@Home and Google Image Game both targeted desktop platforms. Seti@Home was an automated solution while Google Image Game involved user interaction. Smartphones opened new horizons for crowd sourcing, especially participatory sensing. The authors have participated in the development of the mobile application of Idökép, a Hungarian weather service. Idökép had more than a hundred deployed online weather stations and a web portal where users could

input their weather observations and weather photos. The mobile application offered the same possibility. In the month following the release user contributions have doubled. Almost every uploaded photo now comes from mobile devices.

While it's easy to find proposed applications for participatory sensing, academic articles discussing the challenges of such applications and their effects on society, there is merely a couple of successfully deployed applications. Mobile broadband penetration is 74.8 in the developed world while 19.8 in the developing world. While this equals in a larger number of subscribers in the developing regions, most application downloads come from the developed countries. Hence, most crowd sourced applications address "first world problems".

A. Transportation

Getting from point A to point B raises many issues, some of them can be solved by a crowd sourced approach. When using a car, knowing the traffic conditions in advance gives a huge advantage. Waze is one of the many traffic aware navigation applications that uses phone's sensors to identify traffic conditions. When arriving to the destination finding a parking spot will become the main concern. Google launched Open Spot, an experimental application that relied on user interaction to detect freed up parking spaces. The application was a failure, the problem remains unsolved.

Public transportation can benefit of crowd sensed information. HotStop Live is an application that provides live transit information gathered by the users.

III. USE CASES AND CHALLENGES

Wikipedia motivation: using results vs

IV. SOLUTIONS

V. CONCLUSIONS

VI. ACKNOWLEDGMENT

<http://mobithinking.com/mobile-marketing-tools/latest-mobile-stats/a#subscribers>
<http://au.businessinsider.com/comparing-app-downloads-across-countries-2013-6>
<http://www.androidauthority.com/google-labs-open-spot-a-useful-application-that-no-one-uses-15186/>
<http://www.waze.com/> <http://live.hopstop.com/>