Mobile Collaborative Cloudless Computing

Nuno Cruz*†
Email: ncruz@deetc.isel.ipl.pt

*Instituto Politécnico de Lisboa
Instituto Superior de Engenharia de Lisboa
ADEETC
†Universidade de Lisboa
Faculdade de Ciências
LaSIGE - Large-Scale Informatics Systems Laboratory

***Abstract*—Although the computational power of mobile devices has been increasing, it is still not enough for some classes of applications. In the present, these applications delegate the computing power burden on servers located on the Internet. This model assumes an always-on Internet connectivity and implies a non-negligible latency.**

The thesis addresses the challenges and contributions posed to the application of a mobile collaborative computing environment concept to wireless networks. The goal is to define a reference architecture for high performance mobile applications. Current work is focused on efficient data dissemination on a highly transitive environment, suitable to many mobile applications and also to the reputation and incentive system available on this mobile collaborative computing environment. For this we are improving our already published reputation/incentive algorithm with knowledge from the usage pattern from the eduroam wireless network in the Lisbon area.

I. Introduction

Recently, we have assisted to a massive expansion of the computational power and memory of mobile devices (smartphones, tablets, laptops, etc.). However, the current usage pattern of these devices suggest that their CPUs are idle most of the time. A similar pattern, identified in desktop computers usage, lead to the emergence of some projects (e.g. Boinc) that use idle CPU time to speed up large computations.

The opportunities for cooperation between devices in proximity increase in line with the computing power available at each mobile device. Cooperation can foster a multitude of high-level location dependent services, such as recommendation systems and traffic alerts or low level network functions like packet routing. Independently of its application, constraining the traffic to some location contributes to alleviate the load on the wireless infrastructure, reduces costs and give to the user an increased sense of privacy as his present location is not being forwarded to a server somewhere on the cloud. In this paper, we use the term Mobile Collaborative Computing Environment (MCCE) to refer to a distributed and cooperative computing model where mobile devices in proximity cooperate to achieve some (possibly individual) goal. Application examples of MCCE range from providing context information (e.g. road traffic) to distributed computing [1].

Unfortunately, cooperation consumes valuable resources (e.g. battery, memory and CPU time) of the mobile devices. In this paper, we assume that users value their resources and therefore prefer to trade instead of donating them, as opposed to what happens in peer-to-peer file sharing, where resources (mostly bandwidth) have no value to users. In our model, a services market would foster collaboration by allowing users to "sell" resources when they are not particularly keen on their device usage. However, the implementation of a services market requires a long-term memory that allows users to collect their rewards days or weeks later, possibly from a distinct set of devices.

II. Challenges

The system model for the Mobile Collaborative Computing Environment (MCCE) framework assumes that a number of mobile devices are able to communicate using some short range network technology like Wi-Fi or Bluetooth. Participating devices are personal to the user and carry sensitive information that cannot be disclosed to other participants. In addition, participants do not share a common goal.

Naturally, this environment raises additional challenges to the implementations of the MCCE. Its successful deployment depends on the resolution of both technical and social challenges that motivate user's, mobile device manufacturers and wireless infrastructure operators to participate. This section lists the challenges that have been considered as more relevant:

a) Latency: Latency is the key metric that will ultimately dictate the user acceptance of CCE. In the general case, mobile devices should not take longer to perform some action in C3s than in the commercial cloud. To reduce latency any code transfer among devices should be avoided. Computing Blocks are the basic units of computation in MCCEs and provide generic computational functions. Examples are complex arithmetic functions, voice recognition or even some sort of specialized hardware access. A number of computing blocks should be uploaded in advance to the mobile devices from some trusted location (e.g. OS manufacturers web site, App store).

In addition to performing transformations over the input data, the computing block abstraction encapsulates a number of other services, like shared memory blocks or caches (using read and write operations) or the access to specialized

978-1-4799-2736-4/14/$31.00 ©2014 IEEE

hardware components (e.g. a GPS receiver). Not all devices will carry all computing blocks. Instead, computing blocks will be assigned to devices following some externally defined replication policy that matches the devices capabilities. The upload in advance of computing blocks is a limitation that may prevent all tasks from being completed. However, it is fundamental to ensure not only the scalability required to cope with the limitations of mobile devices, but also to reduce task processing time, save the network bandwidth and power that would be required to transfer the code and ultimately ensure that code running on devices is, by design, downloaded from a trusted source.

b) Power Consumption: The MCCE's concentrates its power consumption effort on the mobile device's network interface and CPU, two of their most power demanding components. The impact on the additional power consumed by the CPU is expected to be partially amortized with the gains achieved by replacing cellular network traffic by short range wireless network protocols.

c) Privacy: Privacy is a sensitive issue for MCCEs, as both the devices being used to perform computations and the data transferred for handling by the computation blocks are likely to include user's personal information. Computing blocks play an important role in preventing personal information leakage for both actors. For devices making their CPU time available for third parties, the risk of having malicious code accessing personal data is minimal giving that code has been downloaded in advance from a trusted third-party. Preventing the devices that execute computing blocks from accessing client's data is challenging. The only anticipated solution is data fragmentation between multiple devices to prevent a rogue device from disclosing the full private information (e.g. an audio recognition application).

It should be noted that even the observation of some user participating in a CCE instance may be considered as the disclosure of personal information given that it records the presence of the user at some location. This is an aspect orthogonal to the previous although it should be equally addressed by CCE. Fortunately, some recent work on anonymity (e.g. [2]) can contribute to address this issue.

d) Cooperation: Mobile devices that take advantage of the MCCE but not make their own resources available, or that provide bogus responses produced with minimal computational effort, must be considered selfish. One node's selfish behaviour can threaten the effective deployment of MCCEs, as it can motivate others to present a similar behaviour. The MCCE must be responsible for detecting and punishing selfish behaviour, by refusing to accept tasks from selfish devices.

III. Current Work

After a first approach at addressing the challenges by defining a MCCE framework to be implemented, the work diverged to address the reputation and incentive system and the problems raised by a highly transient neighbourhood, where nodes have occasional Internet connectivity and therefore can be applied to a broad range of mobile applications. We call this approach a Hybrid Trust and Trade Service (HTnT) [3].

HTnT is a hybrid reputation/digital cash service for leveraging mobile collaborative computing scenarios with a long term memory. The system assumes a Central Trusted Entity, that issues and validates virtual currency and manages user's currency accounts and reputation. CTE has no intervention at transaction time. The interactions between the devices and the CTE are orthogonal to transactions. They are expected to occur at the user's convenience (for example, during device charge cycles). HTnT complements virtual currency with reputation information. Reputation information aims to provide knowledge about users past behaviour, allowing users to make more informed choices on the devices with whom they will cooperate. Two classes of reputation are defined: *global reputation*, managed by the CTE, and short-term *local reputation*, built directly by the devices according to their experience on transactions happening in the intervals of their contacts with the CTE.

To improve HTnT, one of the objectives is to disseminate global reputation information in the most efficient way among all participants on the system. Ensuring that at any point in time a device has access to the global reputation information, either stored locally on the device or on a nearby device. To design and evaluate an algorithm to achieve this objective we need a mobility pattern to be applied during simulation.

We analysed the RADIUS access logs of the eduroam network on the Lisbon Polytechnic Institute to obtain mobility traces from the usage patterns, this work is currently in progress, the first statistical analysis of the data is to be published in PerMoby 2014 [4]. This knowledge will allow us to better change HTnT to reflect current usage patterns of wireless networks, based on real data. We are currently in the process of analysing the obtained data, however the sheer amount of data to analyse poses several challenges were a simple extraction of daily contacts between users takes us several weeks.

IV. Related Work

There is multiple related work for each of the addressed portions of the current work, the following are related to the full MCCE concept. Approaches that extend the computing power available to mobile devices using specialized hardware have been experimented. In a cloudlet [5], access points are extended to bring computing power closer to mobile devices, improving latency by avoiding calls to servers located on the Internet. CloneCloud [6] instead replicates mobile devices on powerful servers on the Internet, using the mobile device only for interfacing with the user. In contrast, we argue that groups of mobile devices could themselves provide these services.

The benefits of ad hoc concentrations of devices to perform distributed computations have been evaluated in the past. [6], [7], [8] all depict architectures where the mobile devices are used in an *ad hoc* topology to perform distributed computations. However, none of the existing proposals manages to

support the collaborative nature of the architecture proposed by CCE.

V. Conclusions

This paper presented an approach named Mobile Collaborative Computing Environment (MCCE) that uses idle CPU cycles of devices in the neighbourhood to achieve the computing power typically provided to these applications externally. This is an appealing concept, which can contribute to alleviate the load on wireless cellular networks, reduce application response time and improve availability. However, the implementation of a MCCE raises a number of non-trivial challenges. These challenges appear mostly from the ad hoc nature of the networking environment, characterized by the lack of trust or interest in cooperating among the participants, unstable connectivity, devices limited resources and latency.

Acknowledgment

This work would not be possible without the invaluable contributions from my advisor Professor Hugo Miranda.

References

[1] Y. Busnel, N. Cruz, D. Gillet, A. Holzer, and H. Miranda, "Reinventing mobile community computing and communication," in *Trust, Security and Privacy in Computing and Communications (TrustCom), 2013 12th IEEE International Conference on*, 2013, pp. 1450–1457.

[2] H. Miranda and L. Rodrigues, "Reputation in anonymous vehicular networks," *Journal of Autonomous and Adaptive Communications Systems*, vol. 3, no. 2, pp. 178–197, 2010.

[3] N. Cruz and H. Miranda, "A hybrid trust and trade service for mobile collaborative computing," in *Next Generation Mobile Apps, Services and Technologies (NGMAST), 2013 Seventh International Conference on*, 2013, pp. 1–6.

[4] N. Cruz, H. Miranda, and P. Ribeiro, "The evolution of user mobility on the eduroam network," in *IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops) 2014*, 2013, to appear.

[5] M. Satyanarayanan, V. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Computing*, vol. 99, no. 1, 2009.

[6] B. G. Chun and P. Maniatis, "Augmented smartphone applications through clone cloud execution," in *Procs. of the 12th Workshop on Hot Topics in Operating Systems (HotOS 2009)*, 2009, p. 8.

[7] D. G. Murray, E. Yoneki, J. Crowcroft, and S. Hand, "The case for crowd computing," in *Procs. of the 2nd ACM SIGCOMM Workshop on Networking, Systems, and Applications on Mobile Handhelds (MobiHeld 2010)*, 2010, pp. 39–44.

[8] C. Borcea, D. Iyer, P. Kang, A. Saxena, and L. Iftode, "Cooperative computing for distributed embedded systems," in *Procs. of the 22nd International Conference on Distributed Computing Systems (ICDCS 2002)*, 2002, pp. 227–236.