Literature review

Co-located mobile media experiences through localized cloud

Zola Mahlaza
zmahlaza@cs.uct.ac.za
Department of Computer Science
University of Cape Town

Abstract

Cloud computing refers to the flexible online data storage services, ranging from passive ones, such as online archiving, to active ones, such as collaboration and social networking[3]. It is undoubtedly a great computing platform from the perspective of a developer, for a user, however, it has a number of faults. It's design is centred around characteristics of the Internet. It's faults arise due to mobile device limitations, weak network signals, location of data center, etc. This survey reports on literature whose goal is localize interactions taking place in the cloud. The ideas and design choices presented in the literature will be taken into consideration when attempting to localize the cloud.

1 Introduction

Cloud computing is a large-scale distributed computing paradigm that is driven by economies of scale, in which a pool of abstracted, virtualized, dynamically-scalable, managed computing power, storage, platforms, and services are delivered on demand to external customers over the Internet[5]. This definition encompasses numerous key attributes. These points are:

1. it is massively scalable[5]
2. can be encapsulated as an abstract entity that delivers different levels of services to customers outside the Cloud[5]
3. it is driven by economies of scale[15]
4. the services can be dynamically configured (via virtualization or other approaches) and delivered on demand.[5]

This definition is informative and makes cloud computing attractive from a developers perspective. However, users of cloud computing services have different needs. A more suitable definition for our purposes focuses on users. To this end, Cachin et al[3] define cloud computing as referring to the flexible online data storage services, ranging from passive ones, such as online archiving, to active ones, such as collaboration and social networking. The key attributes users look for are:

1. File security.
2. Transparency in user's data use by company.

3. Elasticity. Availability of services is essential.
4. Location of data. Control over location of data affects speeds of data transfer.

Cloud computing has had beneficial qualities to various areas such as education, academia, business, and possibly other areas as well. An example of a benefit of cloud computing, provided by Armbrust et al[1], is that it allows developers with novel Internet service ideas to not face the problem of buying hardware or the human expense to operate it for deployment of their services. It is undoubtedly a great computing platform from the perspective of a developer, however, for a user it might have a number of faults. Users do not have control over where their data is stored. In essence, it can be observed that cloud computing has short comings in fostering co-located interactions. This is largely because it's design is centred around characteristics of the Internet. Other restrictions of cloud computing arise due to mobile device limitations, weak network signals, location of data center, and possibly other reasons. In an attempt to bridge this gap, mobile cloud computing has surfaced in the past years. Mobile cloud computing offers a platform for connecting mobile devices to web-based services and each other[10]. The is beneficial because these web-based "[...] services can also be introduced for co-located social interactions" as Mäkitalo et al[10] reveal. The focus on co-located interactions arises from indications that research reports that people want to use mobile device when co-located as means to enrich social interactions[6]. This survey reports on literature whose goal is move the cloud closer to the mobile device, in other words, we will look at literature whose goal is localize interactions taking place in the cloud. In particular, we will focus on:

- Co-located file sharing on mobile devices.
- Co-located collaborations using mobile devices.

The result, hopefully, will be the understanding of current network architectures that support co-located interactions through mobile devices. We also hope to understand the challenges in providing cloud services such document collaboration and file sharing for mobile devices. The notions presented in the literature will be taken into consideration when attempting to localize the cloud.

2 Co-located interactions

Technology plays a vital role in many societies, be it in an implicit or explicit manner. It is then vital to be aware of social distinctions between different geographical regions. There are numerous reasons why distinctions could exist, and an example could be the difference in the availability of infrastructure. These distinctions give rise to differences in the use of technological artefacts. As an example, we have observed online media sharing become popular with the advent of services such as Dropbox, Google+, Google Drive, etc. Walton et al[16] have shown, however, that in places such as Khayelitsha (Cape Town, South Africa), co-located phone use surpassed online sharing. The media stored on the mobile devices became public personae, that is, played the role of social media 'profiles'[16]. This observation, together with the fact that research reports that people want to use mobile device when co-located as means to enrich social interactions[6], means that it is then essential to understand and support co-located interactions. The work done by Reitmaier and Benz

[13] introduces the notions of how one should use (or not use) context, time and identify for co-located interactions. They tested the viability of cloudlets by creating an ad-hoc infrastructure that allowed three devices to connect using bluetooth. The result that they presented was that among 10 participants of ages 18-22, cloudlets could enrich co-located media experiences, however, there were concerns over "privacy and identity management".

3 File sharing

A trend of social media sharing websites which focus primarily on image sharing have emerged since 2008. They are different from social networking websites such Facebook, Twitter, VK[1], etc. They do not use the concepts of friends, followers, or groups. They revolve around a body of users which share a variety of images. These websites include Imgur, 9gag, etc. As mentioned, the core mechanic behind them is the sharing of user-generated images. A large number of these images are Internet memes. An internet meme can be defined as a unit of imitation in the form of an image, video, or website[4]. We will use them as a means to present issues and scenarios which can prompt co-located users to share files. To that end, it our understanding that users of these services are likely to share images with co-located friends because they may be funny or possibly for other reasons. This means that when one creates a cloudlet, an intuitive interface that is quick in file transfer regardless of image size, is crucial. This is because doing so will reduce delays and thus not deter the natural flow of interactions. These websites have also created communities, and like any human communities there artefacts which present human emotions. This means that there exists images which are humorous. However, the humour is sometimes linked to other images (thus forming inside jokes). This presents a scenario where users can understand (or possibly not understand) the inside joke. It is similar to what Ah Kun and Marsden[8] identify as storytelling photo-talk. Storytelling photo-talk occurs in conversations where photos are shown to people who have no prior knowledge of the context in which the photo was taken or made. The opposite of that, is known as reminiscing photo-talk. It is the sharing of images exclusively with the original "capture group". In this context, however, reminiscing photo-talk is the scenario when photos are shown to others who have prior knowledge of the image context thus understand the inside joke. File sharing in a cloud setting implies concurrent access to the same files at times. It is then essential to provide software lock mechanisms when it comes to editing the files. Ah Kun and Marsden[8] refer to these mechanisms as floor control policies. There are numerous ways one would restrict modification on resources which have been accessed concurrently. However, the most appropriate methods are ad-hoc and three-second policies as mentioned by Ah Kun and Marsden[8].

Ad-hoc: This is a policy whereby any user can control a resource at any time. Here, there are no software locks. A social protocol is expected to arise from the users[8].

Three-second: This is a policy whereby control is passed around among the users. A user acquires control and performs an action. When there is no action from the user on a given

[1] Russian social network.

time limit (three seconds), control is passed to another user. A user will require control by attempting to perform an action[8].

Ah Kun and Marsden[8] implemented an application for photo sharing in a co-located scenario. The result of their experiment reveal that when using the ad-hoc access policy, the teasing of friends and the “entertainment” aspects flourished. However, no structural resource access protocol emerged, and the session turned into a game. Unlike the ad-hoc policy, in the three-second policy, the users adapted the policy to meet their needs. It showed no signs of turning chaotic.

The effectiveness of file sharing in a mobile network is dependent on where the files are stored. The are currently two viewpoints of mobile cloud computing with respect to data storage (and data processing) [12]. There is the simple viewpoint and also the mobile device viewpoint. The simple viewpoints refers to when data storage and data processing is done outside the mobile device. Mobile device viewpoint, on the other hand, refers to when both data storage and data processing is done on the mobile device. These viewpoints greatly affect the protocols used to exchange files. An approach for data sharing is the use of a peer-to-peer networks, which implies the mobile device viewpoint. In this case, one is likely to use mechanisms such as IP multicasting for peer-to-peer communication. The issue, however, is that peer-to-peer systems ”generally generate a lot of traffic and require not only the resources of every peer (e.g., CPU, memory, and bandwidth) but also many network resources”[11]. An alternative approach for data sharing is the introduction of a server that will handle most, if not all, file storage and coordination of communication between devices. This approach implies the simple viewpoint. This server will create a network within a small range. A client-server network architecture is not without faults. The obvious problem is that it introduces a single point of failure. This means that when the server is down, the entire network is compromized.

Hayes et al[7] have explored the use of a peer-to-peer system in an ad-hoc network for file sharing on mobile devices. They developed an application that uses a variation of the Gnutella protocol and bluetooth as a means for file transfer. Gnutella is an open, decentralized, peer-to-peer search protocol that is used mainly for files [14]. The reason why bluetooth was attractive to use, as they proclaim, is that it “[..] also provides for security and service discovery as part of the protocol, which application development can leverage rather than re-implement”[7]. Their application has a recommender component that uses profiles to recommend files from locally stored files. The purpose of using a profile is to keep track of the users required files and present them when avaialable from their peers. This is a mechanism they have used to circumvent the issue that mobile connections are likely to be short-lived because mobile device users are likely to move out of range frequently. Hayes et al[7] tested their application in order to determine the length of time required for two devices to automatically establish a connection and transmit a file. The tests were conducted on two laptops using USB (Universal serial bus) bluetooth devices and Rococo’s Impronto bluetooth stack for Linux. The recorded time was an average of 48 seconds. This time was for a file of approximately 3 MB. The tested case consisted of the following actions for two devices, A and B:

- Device A searched for nearby Bluetooth devices.
- Device B was discovered.
- Device A searched for the Gnutella service on Device B and retrieved the details required to connect to this service. Device A then used these de- tails to set-up a serial connection to Device B.
- Device A sent a RecRequest packet requesting that Device B recommend a file using the user profile in the packets payload. This profile was created from the artist ID3 tags of stored files.
- Device B sent back the details of its shared files that matched Device As profile.
- After checking these recommendations against the files already in the shared folder, Device A re- quested one of the recommended files by sending a FileRequest packet.
- Device B sent back 9 FileResponse packets, each containing a 500 KB segment of the requested file.

4 Collaborations

A popular cloud service is online collaborations. This is despite that research has shown that computer-mediated communication decreases group effectiveness, increases time required to complete tasks and decreases member satisfaction as compared to face-to-face groups[2]. An example of a scenario that identifies why mobile device collaboration is essential and the problems which face is is given by Luyten et al[9]. The scenario is that there are a number of individuals who are working on a architectural assignment. These individuals are likely to gather at the construction site to discuss certain logistical matters. Suppose that they discuss building sketches, it is likely that each person will have to add annotations to the sketch itself. Using traditional methods, a single person can be delegated to capture points from people and write them down. The second approach would be to rotate the device (laptop or mobile device) among the parties involved, each person adds annotations and passes the device to the next party. There are key attributes in this scenario to take note of. These are:

1. The meeting is “mobile”, that is, it is held in an arbitrary place and not the office.
2. Device(s) involved for viewing and editing the information space have limited resources in terms of computation and device display size.
3. The users are co-located.

The most obvious issue in collaborations is concurrent access. It is worth noting that distributed access to files can still be handled using software lock mechanisms, that is, ”mechanisms which make sure that data stays consistent during simultaneous access [...]”[9]. The the use of the service in a mobile setting implies that the network architecture needs to support portability. For this reason, one cannot use large immobile hardware. Luyten et al[9] have explored collaborative work for mobile devices using an ad-hoc network. They used two applications for evaluating their research. The two applications are called GeoPlanner and Us-Draw-it. ”GeoPlanner is a distributed application that can run on two PPDs”[9]. It allows users to add images and drawings to a shared information space. Users can then manipulate data on the shared space. As mentioned before, the second application is Us-Draw-it. "It is a distributed drawing program for PPDs"[9]. A major distinction between Us-Draw-it and GeoPlanner, however, is that in Us-Draw-it the images that have been drawn are updated as they are drawn. This means that users can view changes in real time. The results of the research done by Luyten et al[9] reveals that even though co-location affords users the possibility to use verbal communication, users often try and manipulate the same object at the same time (they do not communicate access). This means that an ad-hoc software lock mechanism might not be effective. Luyten et al[9] also explored using peephole displays a way to circumvent the display restrictions of mobile devices such as PDAs and cellphones. Peephole displays are spatially aware displays, in which a position-tracked display provides a window on a bigger virtual display. They are a way of solving the issue of small displays on mobile devices. An observation, however, is that peepholes introduced a delay in the application. This delay was caused by the tracking system.

Another way of providing collaboration services for mobile devices is the use of a central server. The use of a client-server network architecture tries to minimize the traffic in the network as compared to a peer-to-peer system. This biggest issue, however, is that the central server needs to be portable. An example of such is the social device platform (SDP) presented by Mäkitalo et al[10]. Here, the (SDP) clients present a interface to users for mobile collaboration. The server in the SDP architecture has multiple components. These are configurator, controller and orchestrator. These components serve different functions, and we will not discuss the functions here as they do not add information of great value and also as a way to keep the survey succinct. The social device platform solves the main issues such as detecting and tracking devices in proximity, discovering appropriate configurations for devices and "orchestrating the operations executions on the devices"[10]. The major restriction, however, is that the social device platform does not allow the personalization of actions based on the context of the co-located users. This is a major restriction as social interactions between users are largely dictated by the context.

5 Conclusion

Cloudlets can be a fully viable solution for enriching co-located mobile device interactions. The requirement for their viability is that they should provide users with control over their data, that is, data ownership should be transparent. In addition to this, data integrity and security should also be provided. Other issues that guarantee the robustness of cloudlets include making them aware of the context of co-located users, and also use an appropriate network architecture. Appropriate, meaning that the network architecture allows the users to be mobile within a range and not fixed in a specific indoor location, does not have intensive traffic and introduces redundancy, that is, eliminates single points of failure.

References

[1] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz,

Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, and Ion Stoica. A view of cloud computing. *Communications of the ACM*, 53(4):50–58, 2010.

[2] Boris B. Baltes, Marcus W. Dickson, Michael P. Sherman, Cara C. Bauer, and Jacqueline S. LaGanke. Computer-mediated communication and group decision making: A meta-analysis. *Organizational behavior and human decision processes*, 87(1):156–179, 2002.

[3] Christian Cachin, Idit Keidar, and Alexander Shraer. Trusting the cloud. *Acm Sigact News*, 40(2):81–86, 2009.

[4] Richard Dawkins. *The selfish gene*. Oxford university press, 2006.

[5] Ian Foster, Yong Zhao, Ioan Raicu, and Shiyong Lu. Cloud computing and grid computing 360-degree compared. In *Grid Computing Environments Workshop, 2008. GCE'08*, pages 1–10. Ieee, 2008.

[6] Richard Harper, Tim Regan, Shahram Izadi, Kharsim Al Mosawi, Mark Rouncefield, and Simon Rubens. Trafficking: design for the viral exchange of tv content on mobile phones. In *Proceedings of the 9th international conference on Human computer interaction with mobile devices and services*, pages 249–256. ACM, 2007.

[7] Anna Hayes and David Wilson. Peer-to-peer information sharing in a mobile ad hoc environment. In *Mobile Computing Systems and Applications, 2004. WMCSA 2004. Sixth IEEE Workshop on*, pages 154–162. IEEE, 2004.

[8] Leonard M. Ah Kun and Gary Marsden. Co-present photo sharing on mobile devices. In *Proceedings of the 9th international conference on Human computer interaction with mobile devices and services*, pages 277–284. ACM, 2007.

[9] Kris Luyten, Kristof Verpoorten, and Karin Coninx. Ad-hoc co-located collaborative work with mobile devices. In *Proceedings of the 9th international conference on Human computer interaction with mobile devices and services*, pages 507–514. ACM, 2007.

[10] Niko Mkitalo, Jari Pkk, Mikko Raatikainen, Varvana Myllrniemi, Timo Aaltonen, Tapani Leppnen, Tomi Mnnist, and Tommi Mikkonen. Social devices: collaborative co-located interactions in a mobile cloud. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia*, page 10. ACM, 2012.

[11] Tetsuya Oh-ishi, Koji Sakai, Kazuhiro Kikuma, and Akira Kurokawa. Study of the relationship between peer-to-peer systems and ip multicasting. *Communications Magazine, IEEE*, 41(1):80–84, 2003.

[12] S. S. Qureshi, T. Ahmad, K. Rafique, and Shuja ul islam. Mobile cloud computing as future for mobile applications - implementation methods and challenging issues. In *Cloud Computing and Intelligence Systems (CCIS), 2011 IEEE International Conference on*, pages 467–471, 2011. ID: 1.

[13] Thomas Reitmaier, Pierre Benz, and Gary Marsden. Designing and theorizing co-located interactions. In *Proceedings of the SIGCHI Conference on Human Factors in*

Computing Systems, pages 381–390. ACM, 2013.

[14] Matei Ripeanu. Peer-to-peer architecture case study: Gnutella network. In *Peer-to-Peer Computing, 2001. Proceedings. First International Conference on*, pages 99–100. IEEE, 2001.

[15] Joaquim Silvestre. Economies and diseconomies of scale. *The New Palgrave: A Dictionary of Economics*, 2:80–84, 1987.

[16] Marion Walton, Gary Marsden, Silke Hareiter, and Sena Allen. Degrees of sharing: Proximate media sharing and messaging by young people in khayelitsha. In *Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services*, pages 403–412. ACM, 2012.