Review of Cloud Computing Architecture for Social Computing

Vaishali D. Dhale
M.Tech Student
Dept. of Computer Science
P.I.E.T. Nagpur

A. R. Mahajan
Professor & HOD
Dept. of Computer Science
P.I.E.T. Nagpur

ABSTRACT

In the recent trend cloud computing is a new business model and cloud computing architecture is a advance topic. With the increasingly ubiquitous nature of social networks and cloud computing users are starting to explore new ways to interact with and exploit these developing paradigms. Social network are used to reflect real world relationship that allow users to share information and form connection between one another. The structure of a social network is essentially a dynamic virtual organization with inherent trust relationship between friends. This paper presents a review of cloud computing architecture for social computing. The architecture propose leveraging the pre established trust formed through friends relationship within a social network to form a dynamic social cloud enabling friends to share resources within the context of a social network.

General Terms

Architecture of social cloud.

Keywords

Cloud computing, Social computing, Social Network, Cloud Computing Platform and Social Cloud.

1. INTRODUCTION

Cloud computing is an exciting area for research, because of its relative novelty and exploding growth. "Cloud computing", as a term for this Internet based service, was launched by industry giants (e.g. Google, Amazon.com, etc.) in late 2006. It promises to provide on-demand computing power with quick implementation, little maintenance, less IT staff, and consequently lower cost.

As far as we know, cloud computing is a new business model and the cloud computing architecture is the advance topic recently. It offers clear SLAs (Service Level Agreements) and is based on a "pay per use" pricing model. Today, everyone enjoy the innovative search engine or social network application for new Internet services no longer require the large capital outlays in hardware to access those service or the human expense to operate it. Currently, Google is the largest search engine and Facebook is the largest social network in the Software as a Service (SaaS).

The structure of a Social Network is essentially a dynamic virtual organization with inherent trust relationships between friends. We propose using this trust as a foundation for resource (information, hardware, services) sharing in a Social Cloud. Cloud environments typically provide low level abstractions of computation or storage. Computation and Storage Clouds are complementary and act as building blocks from which high level service Clouds and mash-ups can be created. Storage Clouds are often used to extend the capabilities of storage-limited devices such as phones and desktops, and provide transparent access to data from anywhere. There are a large number of commercial Cloud providers such as Amazon EC2/S3, Google App Engine, Microsoft Azure and also many smaller scales open Clouds like Nimbus and Eucalyptus. These Clouds provide access to scalable virtualized resources (computation, storage, applications) through pre-dominantly posted price mechanisms. The paper propose leveraging the pre-established trust formed through friend relationships within a Social network to form a dynamic "Social Cloud", enabling friends to share resources within the context of a Social network. We believe that combining trust relationships with suitable incentive mechanisms (through financial payments or bartering) could provide much more sustainable resource sharing mechanisms. The paper is arranged in five sections. Section 1 presents introduction to the topic. Section 2 presents components of social services computing. Section 3 presents social cloud architecture and working. Section 4 and section 5 presents conclusion and references respectively.

2. COMPONENTS OF SOCIAL SERVICES COMPUTING

Fig. 1 shows our Social Services Computing ecosystem. The ecosystem comprises five basic elements which are shown as follows.

- **Service providers and service consumers**

In an abstract level, service consumers are computing entities which present individual persons, organizations and other entities. Service providers provide computing services for service consumers. A service provider becomes a service consumer when it consumes the services provided by other service providers.

- **Services**

Services have their definition. Services represent a type of relationships-based interactions (activities) between at least one service provider and one service consumer to achieve a certain business or solution objective. In our ecosystem, services are computing interactions between at least one service provider and one service consumer to achieve a certain computing goal.

- **Local services**

Local services are the lowest level services that focus on collecting data or controlling physical things. Local services run on physical things or other computer systems. They interact with higher level services existing in cloud computing platforms.

- **Physical things**

Physical things are embedded devices that power themselves or be supplied power by other systems.

- **Cloud Computing platforms**

Cloud Computing platforms are platforms with Cloud Computing infrastructure which supports Services Computing.

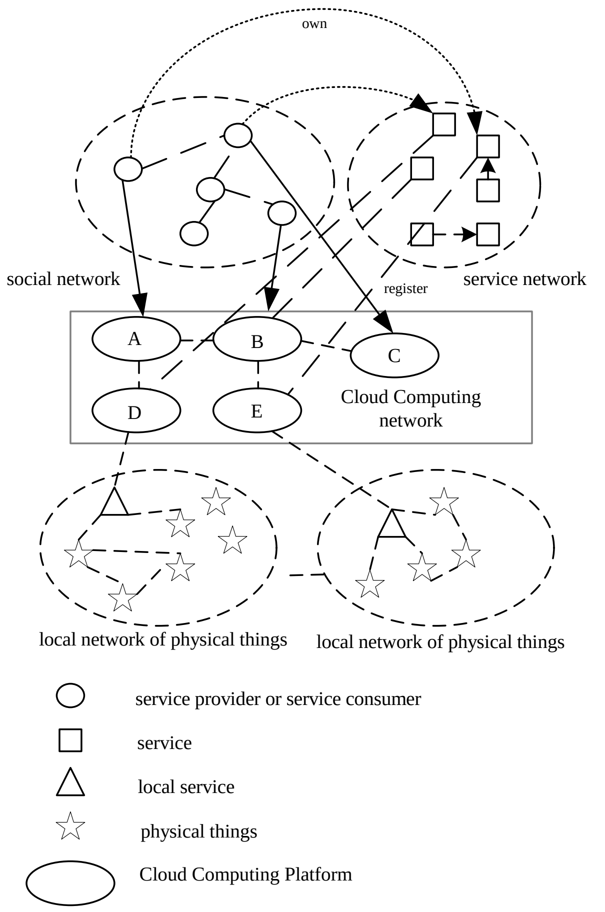

Fig 1: Social Services Computing ecosystem

In Fig. 1, service providers register themselves to distinct social networks that are deployed in different Cloud Computing platforms. Different Cloud Computing platforms are connected together and exchange data according to requirements of real computing tasks. There are four kinds of networks in Fig. 1. They are

- **Social networks**

Service providers and service consumers are nodes in the social networks. The relationships between service providers and service consumers involve friendship, business partnership, and so on.

- **Services networks**

Services may depend on other services, and a service may comprise other services. Hence relationships among services are similar to object relationships in the real world.

- **Cloud Computing networks**

In a Cloud Computing network, nodes interact with each other through data communication channels. Cloud Computing platforms collaborate to conduct computing tasks.

- **Physical thing networks**

Physical things such as wireless sensors are connected together to cooperatively achieve a special computing goal, for example to monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants.

3. SOCIAL CLOUD ARCHITECTURE & WORKING

In a Social Cloud, services can be mapped to particular users through Facebook identification, allowing for the definition of unique policies regarding the interactions between users.
A specialized banking component manages the transfer of credits between users while also storing information relating to current reservations. A high level architecture of a Social Cloud is shown in Fig. 2. This section presents social cloud architecture and their working.

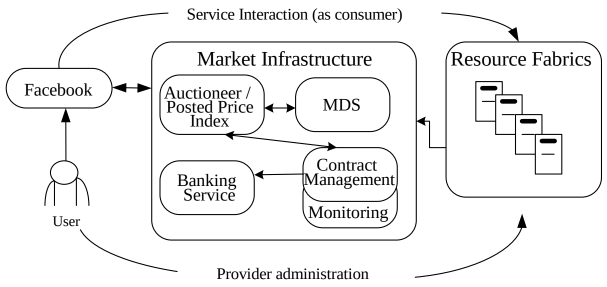

Fig 2: Social Cloud Architecture

3.1 Facebook Applications

Facebook exposes an application API through a RESTlike interface which includes methods to get a range of data including friends, events, groups, application users, profile information, and photos. Facebook Markup Language (FBML) includes a subset of HTML with proprietary extensions that enables the creation of applications that integrate completely with the Facebook look and feel. Facebook JavaScript (FBJS) is Facebook's version of JavaScript – rather than sandboxing JavaScript, FBJS is parsed when a page is loaded to create a virtual application scope. Facebook applications are hosted independently and are not hosted within the Facebook environment. A Facebook canvas URL is created for user access, this URL maps to a user defined callback URL which is hosted remotely. The process of rendering an application page is shown in Fig. 3. When a page is requested by the user through the Facebook Canvas URL (http://apps.facebook.com/socialcloud/) the Facebook server forwards the request to the defined callback

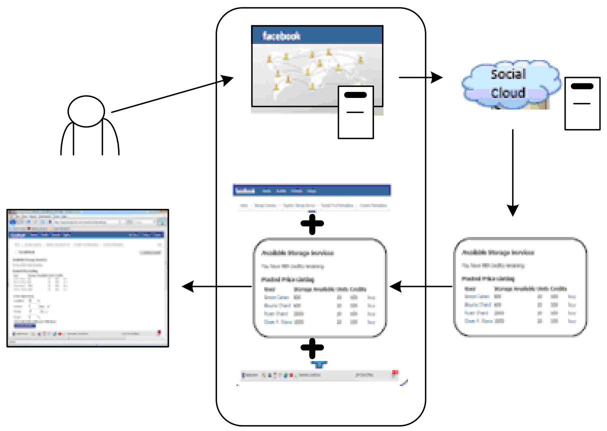

Fig 3: Facebook application hosting environment

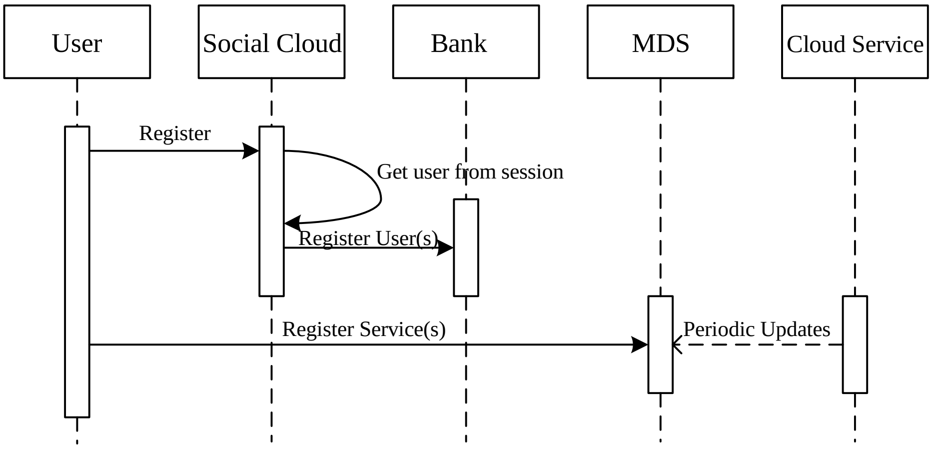

Fig 4: Registration in a Social Cloud

URL. The application creates a page based on the request and returns it to Facebook. At this point the page is parsed and Facebook specific content is added according to the FBML page instructions. The final page is then returned to the user. This routing structure presents an important design consideration in a Social Cloud context as access to the Cloud services would be expensive if routed through both the Facebook server and the callback application server in order to get data from the actual Cloud service. To reduce this effect FBJS can be used to request data asynchronously from the specified service in a transparent manner without routing through the application server.

3.2 Virtualized resources

Cloud computing relies on exposing virtualized resources as a service in a metered and elastic manner. A Social Cloud service could represent any resource that users may wish to share, ranging from low level computation or storage through to high level mash-ups such as photo storage. There are two generic requirements of this service: firstly, the interface needs to provide a mechanism to create a stateful instance for a reservation. In our model the Social Cloud application passes a SLA to the service which is parsed and used to instantiate the required state. Secondly, in order to be discovered the service needs to advertise capacity so that it can be included in the market. In our design this advertised capacity is XML based metadata which is periodically refreshed and stored in Globus Monitoring and Discovery System (MDS).

3.3 Banking

The prototype Social Cloud includes a credit-based system that rewards users for contributing resources and charges users for consuming resources. The banking service registers every member of the cloud and stores their credit balance and all agreements they are participating (or have participated) in. Credits are exchanged between users when an agreement is made, prior to the service being used. To bootstrap participation in the Social Cloud, users are given an initial number of credits when joining the Cloud. While suitable for testing, this initial credit policy is susceptible to inflation and cheating (if fake users are created and the initial credits are transferred). Currently there is no mapping between Social Cloud credits and real currencies or Facebook credits.

3.4 Registration

Fig. 4 shows the registration process. – users first need to register themselves, and then specify the Cloud services they are willing to trade. As users are pre-authenticated through Facebook, user instances can be transparently created in the banking service using the users Facebook ID. Having registered, the user is presented with an MDS End Point Reference (EPR) and Cloud ID which they use to configure their service for registration (and refreshment) of resource capacity. Market services utilize the MDS XPath interface to discover suitable services based on user IDs and real time capacity.

3.5 Service Marketplaces

Service usage is exchanged for credits within a marketplace. The Social Cloud marketplace is generic and not limited to a specific type of market, although two implementations are provided.

3.5.1 Posted Price

In a posted price model providers advertise offers relating to particular service levels for a predefined price or following a linear pricing function; consumers are then able to fine tune specific parameters to create a SLA. Creating such a market requires coordination between a numbers of the Social Cloud components to; discover Cloud services, create agreements, and transfer credits. Fig. 5 shows the flow of events for a posted price trade in a Social Cloud. When a user requests posted price offers the Cloud application uses the user ID (from the session) to check the user is registered in the bank and they have sufficient credits available. A list of all the users' friends is generated using the Facebook REST API, this list is used to compose a query to discover particular Cloud services from MDS. The result of which populates the offer list that describes availability and pricing information. When the user selects a Cloud service, the Social Cloud application creates a SLA which it sends to the Cloud Service. Assuming both parties accept the agreement it is then passed to the Bank to transfer credits between users.

3.5.2 Auctions

In an auction-based market trades are established through a competitive bidding process between users or services. Like the posted price market, a list of friends is discovered and passed to a specialized auctioneer to create and run the auction. Fig. 6 details the auction process. In this example a reverse auction protocol is used, where Cloud services

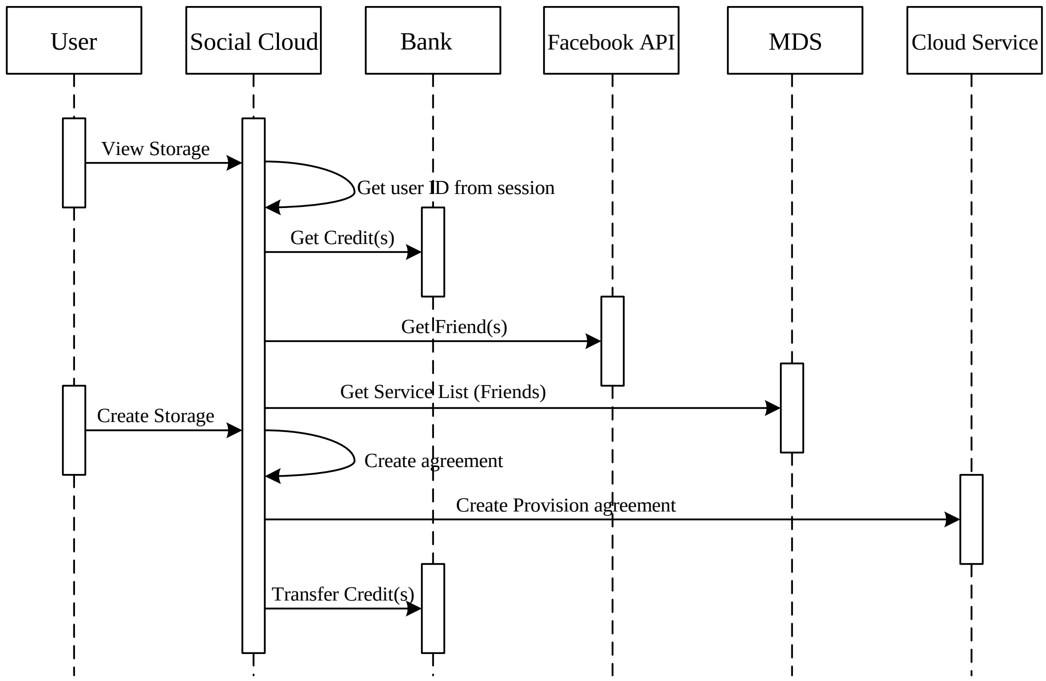

Fig 5: Posted Price marketplace in a Social Cloud

compete (bid) for the right to host the user's task. The auctioneer uses the list of friends to locate a group of suitable Cloud services; these are termed the bidders in the auction. Each provider requires an agent to act on its behalf to value resource requests, determine a bid based on locally defined policies, and follow the auction protocol. The auctioneer determines the auction winner and creates a SLA between the auction initiator and the winning bidder. As in the posted price mechanism, the agreement is sent to the specified service for instantiation and the bank for credit transfer.

4. CONCLUSION

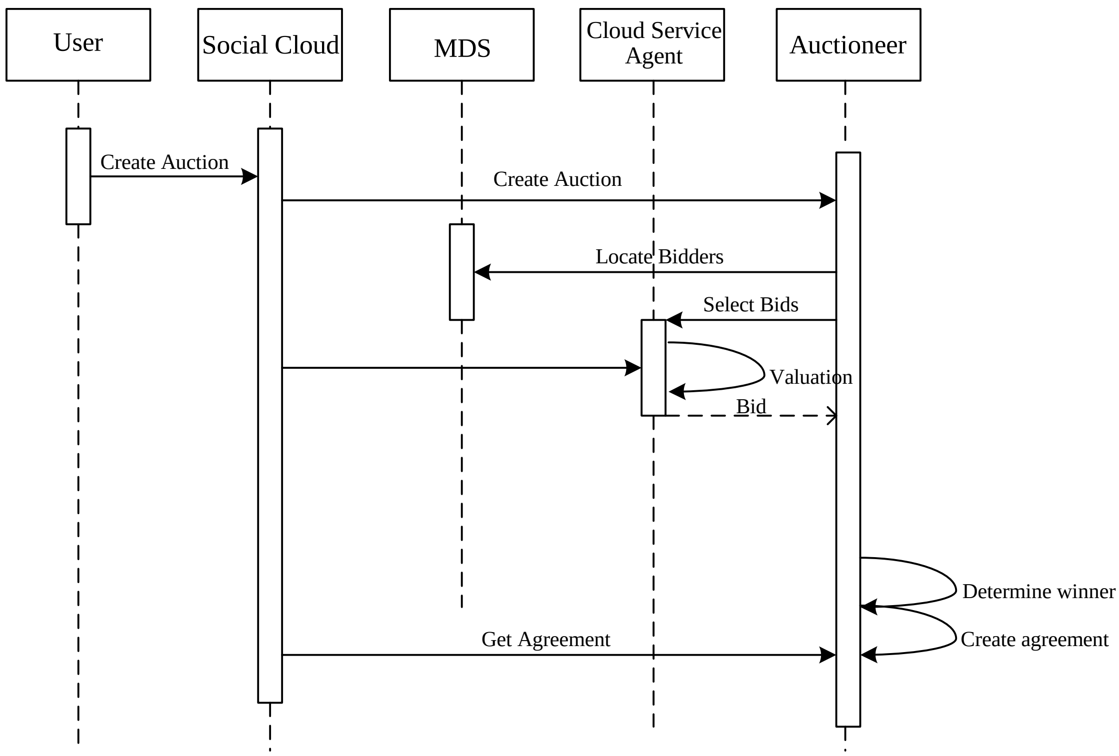

Fig 6: Auction marketplace in a Social Cloud

This paper has presented the architecture of a social cloud an amalgamation of cloud computing, volunteer computing and social networking. In our system facebook users can discover and trade storage services contributed by their friends taking advantages of pre existing trust relationship. Our future work aims to generalize our approach so that we can capture additional marketplaces.

5. REFERENCES

[1] Bo-Wen Yang, Wen-Chih Tsai, An-Pin Chen and Singh Ramandeep, "Cloud Computing Architecture for Social Computing - A Comparison Study of Facebook and Google" In IEEE International Conference on Advances in Social Networks Analysis and Mining, pp. 741-745, 2011.

[2] Suke Li and Zhong Chen, "Social Services Computing: Concepts, Research Challenges, and Directions" IEEE/ACM International Conference on Green Computing and Communications & 2010 IEEE/ACM International Conference on Cyber, Physical and Social Computing, pp. 840-845, 2010.

[3] "Cloud computing, issues, research and implementations." ITI 2008 30th International Conference on Information Technology Interfaces, vol. 4, no. 4, pp. 31-40, 2008.

[4] B. Peng, B. Cui and X. Li, "Implementation issues of a cloud computing platform," IEEE Data Eng. Bull., vol. 32, no. 1, pp. 59-66, 2009.

[5] Nimbus: Open Source Infrastructure-as-a-Service Cloud Computing Software, 2009.

[6] Daniel Nurmi, Rich Wolski, Chris Grzegorczyk, Graziano Obertelli, Sunil Soman, Lamia Youseff, and Dmitrii Zagorodnov. The eucalyptus open-source cloud-computing system. In Proceedings of 9th IEEE International Symposium on Cluster Computing and the Grid (CCGrid 09), Shanghai, China. 2009.

[7] M.E.J. Newman and J. Park. Why social networks are different from other types of networks. Phys. Rev. E, 68:036122, 2003.

[8] Josep M. Pujol, Vijay Erramilli, and Pablo Rodriguez. Divide and conquer: Partitioning online social networks. CoRR, abs/0905.4918, 2009.

[9] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia,Above the Clouds: A Berkeley View of Cloud Computing, February 10, 2009.

[10] Facebook. Cassandra. http://www.facebook.com/note.php?noteid=24413138919