VIETNAM NATIONAL UNIVERSITY - HO CHI MINH CITY

INTERNATIONAL UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

A Service On Decentralized Online Social Networking

by

Advisor: MSc. Vo Duy Khoi

Student: Le Quoc Thanh

A pre-Thesis report submitted to the School of Computer Science and Engineering in partial fulfillment of the requirements for the degree of Bachelor of Computer Science and Engineering

Ho Chi Minh City, Vietnam 2013

Contents

Chapter 1 3
I - Motivation scenario of your report 3
II - Problem statement 3
III- Scope of your thesis: 4
Chapter 2 5
I – Similar system review: 5
II - Technology review 7
1) Post: 7
2) Chat: 10
Chapter 3 11
I - Requirements of your problem: 11
II - Analysis of your problem: 11
III - Design of your problem 13
Chapter 4 15
I - Time to do the rest of job: 15

Chapter 1

I - Motivation scenario of your report

Nowadays, online social networking is an important part of millions Web users' everyday online activities with some of the popular and biggest websites, which are known as Facebook, Twitter, MySpace. Online social networking applications have greatly improved our ways in interacting with friends in the digital life. Moreover, many nested services of that application such as post, chat, fan page help us to keep in touch with existing friends or to make new ones. However, the currently popular online social networking is centralized, which means they store all information that people are generating or consuming. Therefore, users have less control over their own data and their privacy information and have to depend on Online Social Networking (OSNs) providers. Hence, this issue of control and privacy in centralized Online Social Networking has motivated my research with a proposal a very different alternative model as decentralized OSNs where the data of users are kept in their own clients.

II - Problem statement

- Actually, for Online Social Networking Websites present problems such as:

+ Such sites do not allow users much control over how their personal information is presented to their online friends.

+ The companies providing the services have the sole authority to control all the data of the users, which results in potential privacy problems.

+ Users have to agree with the policies of the social

networking sites when using their services, even though they may involve in usage of their data for targeted advertising.

- However, Decentralized Online Social Networks (DOSNs), which can solve these problems by three main respects:

- Privacy: Users in decentralized social networks decide who to show the information to and what restriction there is on the data.
- Ownership: As the information is stored on a trusted server or on the local computer, users have complete ownership of the data. They would not lose their data suddenly as the proprietary service hosting their data decides to shut down without much notice
- Dissemination: Information is disseminated according to users' preferences and friendship relations.

III- Scope of your thesis:

This paper is focused research and proposes a demo service of DOSNs application on mobile as post service to prove that DOSNs have the potential to provide a better environment within users communication. In addition, users can have more control over their privacy, and the ownership and dissemination of their information, which can solve existing problems of centralized system. More importantly, a decentralized approach to OSNs breaks the boundaries between social networking sites by providing users more freedom to interact with each other.

Chapter 2

I – Similar system review:

- Decentralized Online Social Networking is trend of research in many companies that interest in social networking field. So, there are some existing applications presented below:

- LifeSocial: primarily aims at keeping social networking services scalable by distributing the load to their users' resources. It is designed with the main premise to leverage on existing and proven components and to create a modular plugin-architecture to assure extensibility.

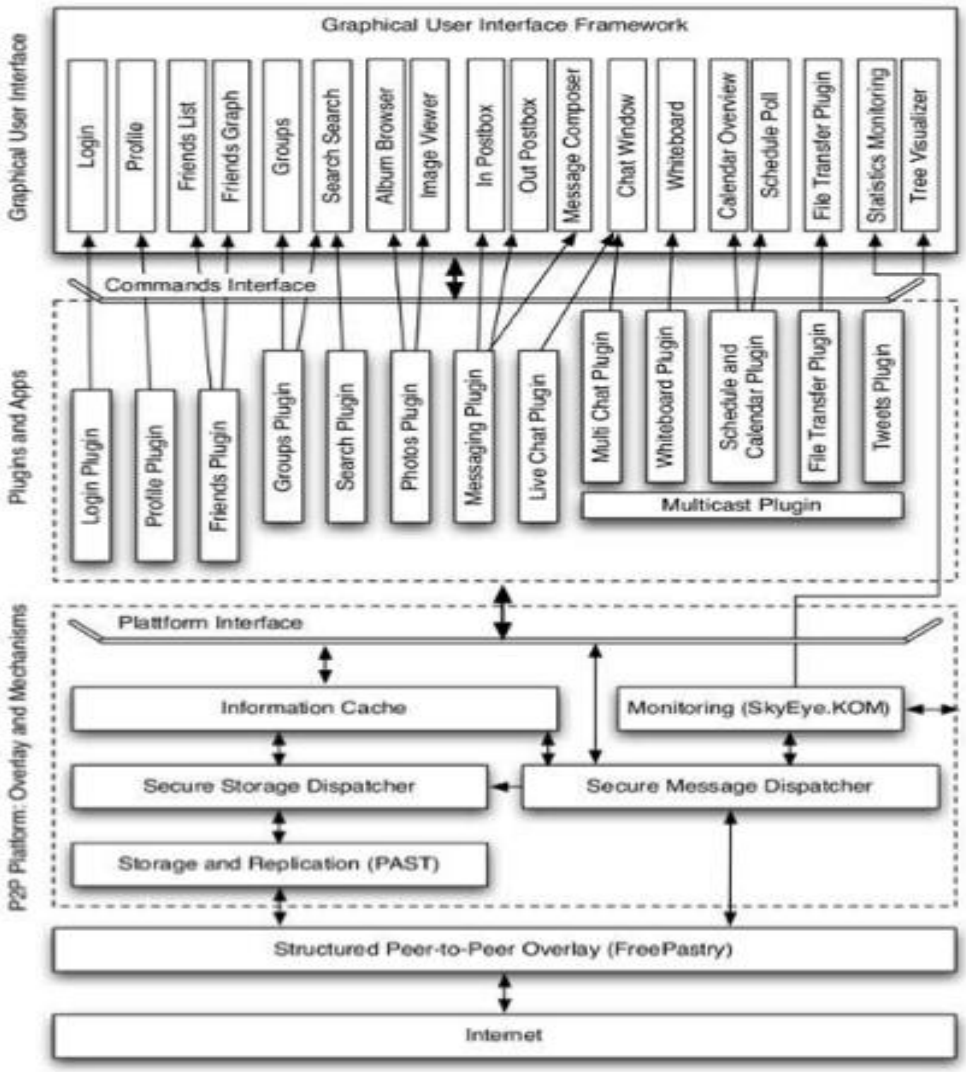

Fig. 4. LifeSocial Plugin Architecture

- Safebook: main objective of protecting its users' privacy, integrity, and availability. Moreover, Safebook consists of three major different components, the TIS, matryoshkas, and a peer-to-peer location substrate.
- FOAF: The framework enables users to export their FOAF profiles, store them on dedicated trusted servers. Users query and manage these profiles through open Web-based protocols such as WebDAV or SPARQL/Update

- PeerSoN: The main properties of PeerSoN are encryption, decentralization, and direct data exchange, which aims at keeping the features of OSNs but overcoming two limitations: privacy issues and the requirement of Internet connectivity for all transactions. To address the privacy problem, it uses encryption and access control coupled with a peer-to-peer approach to replace the centralized authority of classical OSNs

- Likir: aimed to protect the overlay against attacks common to these systems, by embedding a strong identity notion at overlay level. And the main motivation of Likir is to avoid a central data repository, for both the reason to avoid a single point of failure and aggregation of user data.

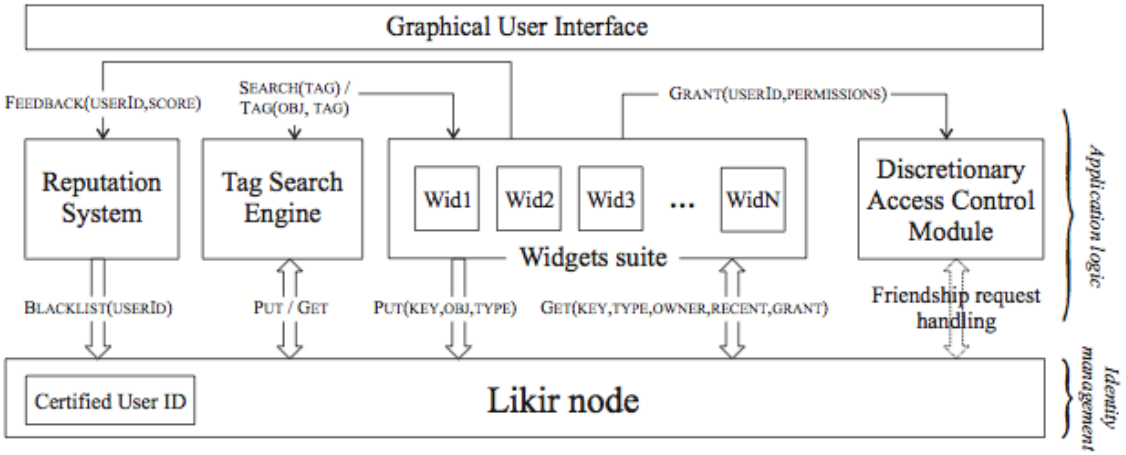

Fig. 5. Likir Architecture

II - Technology review

- This paper research some technology used in two services on OSNs as post and chat of the biggest website about OSNs, Facebook.

1) Post: they used pull-push technology for this service. In more detail, we review some background information about this technology.

- Push technology is the ability to transfer information as a reaction into event occurrence. Hence, data can be sent to users without having them specifically ask for it. The advantage of push is straightforward.

- Push-based: when a request is received at a server, the information of interest and is located by the server and returned to the clients. This request-response style of operation is pull-based - the transfer of information from servers into clients is initiated by a client pull.

- Pull-based: Push-based data delivery involves in sending information to a population of clients in advance of any specific request. With push- based delivery, the transfer is initiated by the server.

- The way to data delivery:

+ First, push is just one dimension for the larger design source of data delivery mechanisms. Second, networked information system employs different data delivery options between different sets of information producers and consumers. We have identified three main characteristics that can be used to compare data delivery mechanisms: (1) push vs. pull; (2) periodic vs. aperiodic; and (3) Unicast vs. 1–to-N. We have found that these three characteristics provided a good initial basis for discussing about many popular approaches.

+ In particular, we argue that all these three characteristics must be considered to make the intelligent choices about delivery

mechanisms for specific situations.

+ The figure 1 below shows those characteristics and the relationship between them and several common mechanisms.

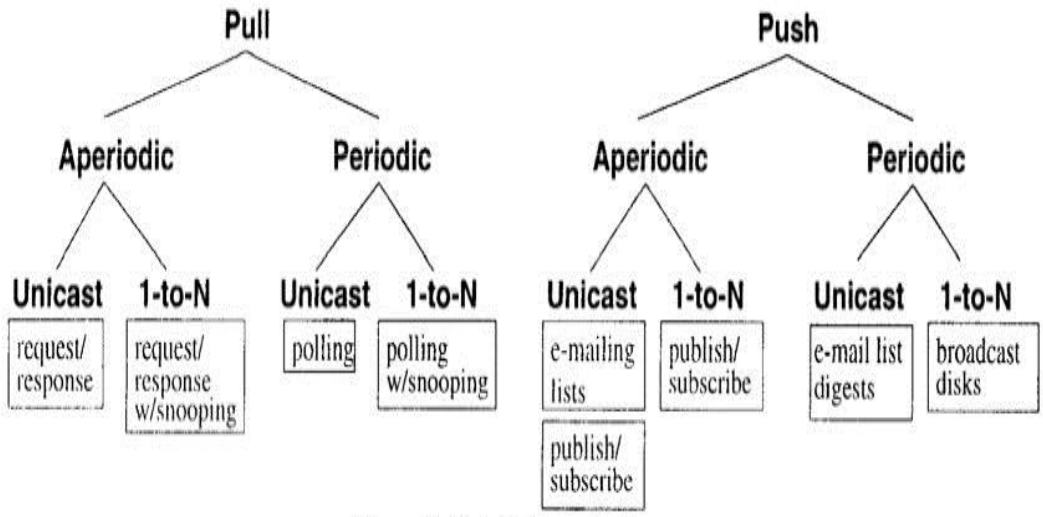

Figure 1: Data Delivery Options

+ Aperiodic delivery: is event-driven-a data request (for pull) or transmission (for push) is triggered by an event such as a user action (for pull) or data update (for push).

+ Periodic delivery: is performed according to some prearranged schedules. These schedules may be fixed or generated by some degrees of randomness.

+ Unicast communication: is the data items sent from a data source (e.g., a single server) to one other machine.

+ 1-to-N communication: allows multiple machines to receive the sent data by a data source. Two types of 1-to-N data delivery can be distinguished:

- Multicast: data is sent to a specific subset of clients who have indicated their interests in receiving the data.

Network proto can be developed, which guarantees the eventual delivery of the message to all clients who should receive it.

- Broadcasting sends information over a medium, which an unidentified and possibly unbounded set of clients can listen to.

+ Aperiodic Pull: Traditional request/response mechanisms use aperiodic pull over a unicast connection. If a 1-to-N connection is used instead, then clients can "snoop" on the requests made by other clients, and obtain data that they have not explicitly asked for.

+ Periodic Pull: a system may periodically send requests to other sites to obtain status information or to detect the changed values. Most existing Web or Internet -based "push" systems are actually implemented using Periodic Pull between the client machines and the data sources.

+ Aperiodic Push: Publish/subscribe protocols are becoming a popular way to disseminate information in a network. In a publish/subscribe system, users provide information, which indicates the types of information they wish to receive. Publish/subscribe is push-based; data flow is initiated by the data sources, and is aperiodic, as there is no predefined schedule for sending data. Publish/subscribe protocols are inherently I-to-N in nature. However, due to limitations in current Internet technology, they are often implemented using individual unicast messages to multiple clients.

+ Periodic Push: has been used for data dissemination in many systems. An example of Periodic Push using unicast is Internet mailing lists that send out "digests" on a regular schedule

2) Chat:

- Facebook chat works in the same way as most of the other Instant Messengers. It uses some technology called AJAX, Jabber (XMPP), Comet, and JSON.

+ For AJAX: it is a method of web development that helps to create interactive web applications, without refreshing the web page. Also, Comet is a web application model that opens a long running HTTP request allows a web server to push data to a browser while browser need not request to server.

+ About JSON: it is just a way to present data in Javascript.

+ Jabber: helps Facebook Chat available everywhere, which is an open messaging protocol supported by most of the instant messaging software, including iChat, Pidgin, Adium, Miranda and more.

- For more details about sending, receiving message, and buddy list:

- Sending message: it will send a POST request in term of

http://www.facebook.com/ajax/chat/send.php?msg_text=hey&msg_id=1234567890&to=FRIENDSID&client_time=1209558664256&post_form_id=297fdbad61d3b1d88f0c311cda25bbbc

With msg_text is the text of the message; msg_id is a random unique number id for messages; client_time is the current time in microseconds; post_form_id used to be known as a hidden form value on any Facebook pages.

- Receiving message: Facebook uses Comet and JSON approach to receive incoming messages.
- Buddy List: The list of your friends who are online is

polled about every 3 minutes with a POST request as http://www.facebook.com/ajax/presence/update.php?buddy_list=1

Chapter 3

I - Requirements of your problem:

When centralized OSNs services change to DOSNs, it is exist many challenges that need to be resolved before building a service on mobile platform. So, some problems can be showed in term of question following here:

+ How to two friends using mobiles (nodes) can find each other?

+ How to consider one node to become a server node and temporary server node when the main one is downed?

+ How to one node can authenticate with centralized social network (Facebook) based on server node?

+ How to one node write a status and their friends see that immediately at their news feed page?

+ Where will user's status be stored?

+ How to synchronize user's data on mobile to social network web platform?

As we see that, these questions above are just some of problems when we change from centralized OSNs to decentralized OSNs. Hence, in this paper, we will research to find out and propose as many suitable solutions as possible to solve them.

II - Analysis of your problem:

- Based on some problems presented above, there are some proposal solutions for each problem that we can consider in the thesis scope.

+ P2P mobile framework will help not only two mobiles can find each other, but also many nodes in different location. In thesis paper, we will go to more detail this technique.

+ Considering one node to become a server node is quite complicated. It based on location and number of connection between nodes. In this field, we can use P2P mobile framework in link recommendation.

+ For authenticate one node to centralized OSNs as Facebook, we can use OpenID to implement this function. It is supported by almost big social networking website.

+ Post a new status is the main field in my thesis. For this field, I propose to use pull-push technology that presented in chapter 2 above. In specify, one node writes a status in their wall-post, push technology will push the post to server node. After that, their friends will pull data through out a request to server node. In addition, we will consider some factor when user pull data instead of pull all of data of all friends, because many information from stranger friends are not all interesting to the user. Therefore, for each node, the following data will be recorded in the simulation:

- Relationship model established between each node with its friends.
- Interest level of each node.
- The number of social data a node shared with its friends.
- The number of social data filtered by relationship model in each node.

- The number of social data that is forwarded by a node.
- The total number of nodes that forwarded social data received from its friends and interest level at the moment of forwarding.

+ SQLite will help to store user's social data on mobile. It is an open source and very popular.

+ A synchronized mechanism between mobile and web platform need time to be researched. In thesis paper, I will try to present that.

- In my point of view, these solutions are general information to help us solving problem. In thesis scope, we will present more detail and build a demo service DOSNs on mobile.

III - Design of your problem

- Following is the simple chart that presents general information how to post service active in decentralized online social networking.

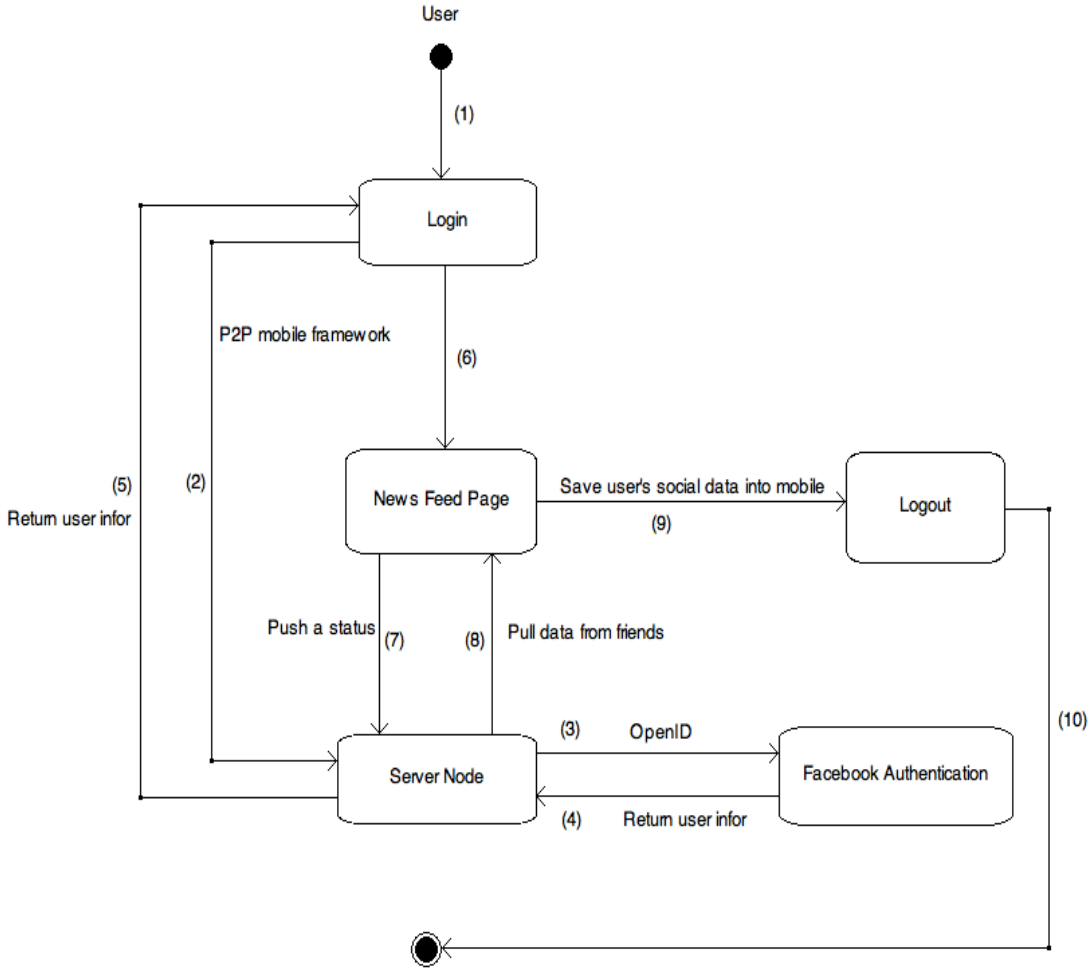

Fig 1. Activity diagram for post service on Decentralized OSNs.

- Scenario description:

1) User open application and login account.

2) Application use P2P mobile framework to find the server node to send account information and ask for login to centralized social networking.

3) Server node receives user account information and use OpenID technology to login to centralized social networking.

4) Centralized social networking return user information if login successful, vice versa it returns fail.

5) If login successful, server node will return user information to user node, vice versa it returns fail and user node need to login again.

6) After login successful, user interface of user node will go to news feed page in which user will see other posts from their friends. In addition, they can write a status in that page also.
7) When they finish writing a status, push technology will active to push the post to server node.
8) In the user home page, they can pull data by a request to server node.
9) When user shutdown the application, all of user's own data will be stored to SQLite at their mobile.
10) Logout or shutdown application.

Chapter 4

I - Time to do the rest of job:

- Plan for the job in thesis scope:

+ Firstly, Researching P2P mobile framework and apply for post service on DOSNs.

+ Second, researching pull-push technology to build a demo application for post service on DOSNs.

+ Thirdly, find out synchronized mechanism between mobile and web platform.

+ Finally, build a demo DOSNs application running on mobile.

- In the future, I will continue to implement this application to release version, which helps users control their social data easier and more safety.