

MySocials Gallery: Application for Browsing Galleries of Images from Social Networks

Anna Samoryadova, Andrew Bogachev, Pavel Shiryaev, Kirill Kulakov
Petrozavodsk State University
Petrozavodsk, Lenin st., 33, Russia
{samoryad, bogachev, shiryaev, kulakov}@cs.karelia.ru

Abstract

Social networks are the most popular and salable services all over the world. People can share their images via these services. One person could have several accounts on different services and different data uploaded to them. Therefore, there is a need of special application for working with users' images from several services at the same time. In this paper we describe the current state of FRUCT R&D project MySocials from Petrozavodsk State University and its part MySocials Gallery, an application for browsing images from various social networks. Current version of the application supports such services as VKontakte, Facebook, Flickr and MyWorld from mail.ru.

Index Terms: social network, gallery, image, browsing, mysocials

I. INTRODUCTION

Nowadays people prefer to be in touch with one another and meet new friends in the Internet. The easiest way to do it is via various social services. One of the most important functions provided by social services is an opportunity to exchange different data between social group members. We have to mention image as the most popular media type which is published in the Internet. There are also special services based on photo sharing called photo hostings, e.g. Flickr [1].

An active user of the Internet usually has several accounts on different social services. Sometimes it is hard to be updated about all the changes that happen every day. Therefore, there is a need of special application which could combine working with different services into working with one client.

One of the most popular ways to share some information with other users is to download your photos or videos. If you have many friends in different social networks you probably want to follow all the changes in their albums or galleries. This problem can be solved by using only one application, which provides an opportunity to browse all the pictures from several services simultaneously.

This paper is a progress report about MySocials project, social network aggregation service. Particularly this paper is dedicated to MySocials Gallery application which is a part of MySocials project. MySocials Gallery is a client for working with images from social services such as VKontakte [2], Facebook [3], Flickr and MyWorld from mail.ru [4].

The project was initiated at Petrozavodsk State University (PetrSU) FRUCT laboratory of wireless and mobile technologies [5]. MySocials belongs to the family of FRUCT research projects [6].

The rest of the paper is organized as follows. Section II describes the idea of MySocials project. Section III demonstrates the architecture of MySocials project. Section IV contains detailed description of MySocials Gallery architecture. Section V describes the implementation of MySocials Gallery. Section VI concludes the paper.

II. IDEA OF MYSOCIALS PROJECT

Nowadays social networks are the most popular and salable services. There are several world leaders and some strong regional services in this area. Also there are universal and special social networks. For example, Facebook and VKontakte are the universal social networks, but Flickr and Picasa Web Albums [8] specialize in images and videos. Many users have accounts in different social services at the same time. It could cause some practical inconveniences and addition waste of time for users.

MySocials projects group (MySocials group) solves these problems by creating a group of integrated services that provide permanent access to the most popular social services from your mobile or desktop platforms. The MySocials idea is to make the work with different social networks easier and provide the integration with some platform key services. Using several applications and plugins user can access to similar functionality of different social networks. MySocials group contains standalone applications and several plugins for different frameworks.

MySocials is a client of social network. It combines the most of functionality from other MySocials projects. *Telepathy VKontakte* is a Telepathy service plugin to access to VKontakte service using API. *QMF Socials* is a set of plugins for Qt Messaging Framework. These plugins allow the user to receive and send messages from QMF based applications. *Sharing plugin for VKontakte service* is a plugin for standard image viewer on Maemo 5 platform. This plugin provides an opportunity to send your image from mobile device to the service in one click.

MySocials Gallery is also a part of MySocials group. It's an application for browsing images from different social services. This application has following advantages: convenient browsing of albums and images from social networks and ability to browse those pictures that you saw earlier even if you have no network connection on your device. You can see comments to the images and send your own comment using MySocials Gallery application.

III. ARCHITECTURE OF MYSOCIALS PROJECT

There are several common parts in projects from MySocials group. For example, projects for VKontakte service use identical API to access data. Therefore, architecture could be divided into four base layers (see Figure 1).

Network layer includes used social services and environment for data transmission. The architecture model contains social networks (e.g. VKontakte, Facebook and Flickr). Every service has its own data transmission protocol. MySocials group supports protocols based on XML and JSON formats.

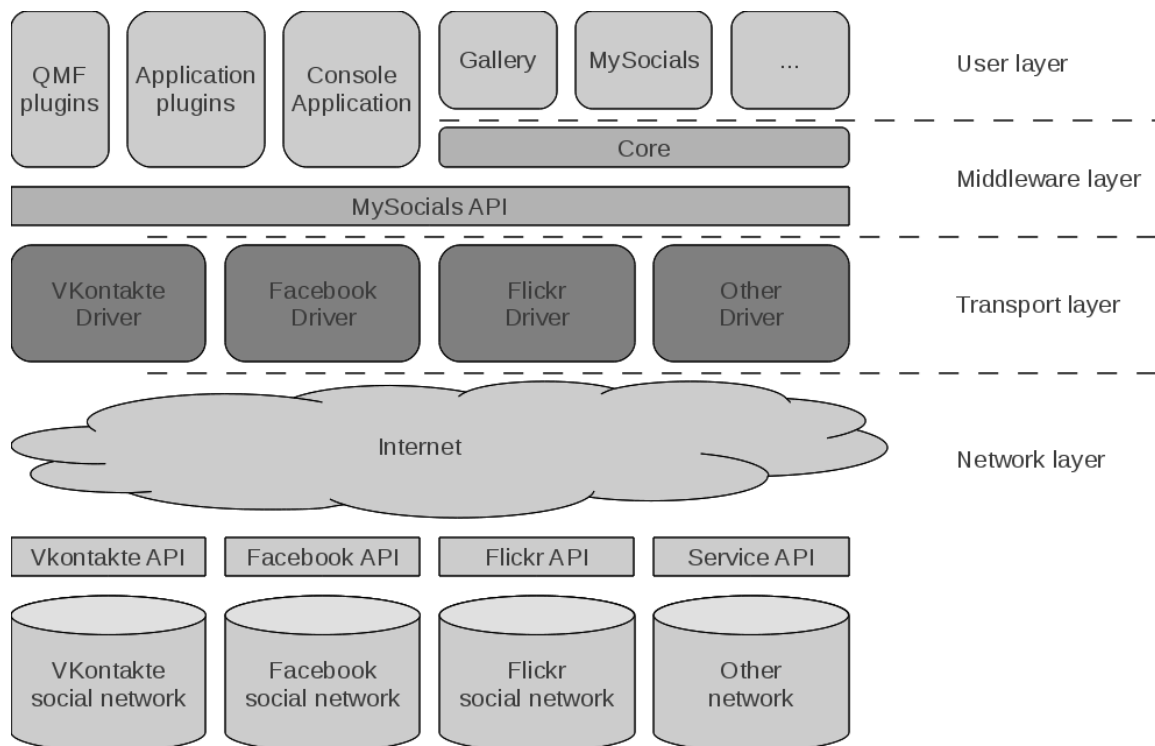


Fig. 1. Architecture of MySocials group

Transport layer is a set of libraries for access to social networks (drivers). The main goal of each driver is to perform requests to the service for getting or sending data. One request to the driver can be implemented as a sequence of requests to the service.

Middleware layer provides common programming interface (MySocials API [7]) for access to different social services. MySocials API is based on XML format. Core is a special Qt library for developing applications which work with different social services. It includes a set of base classes, tools for caching data for autonomous work of application and compound requests. Unification of all base functions in the core makes implementation of different MySocials services easier and helps to avoid equal parts of code.

User layer includes various plugins and services which were implemented during the work on MySocials group.

IV. MYSOCIALS GALLERY ARCHITECTURE

Every social service which allows users to share their photos or other images has a definite set of functions and services. Usually images are distributed into several albums. Each image and album contains associated information tags (e.g. creation time, uploading time, etc). The functions include working with albums and images, comments and tags to them. However, each service has its own unique features. We have to consider some of them before we start to describe MySocials Gallery application.

First of all, let us view the most popular multi-services – social networks such as Facebook and VKontakte. These services allow to use different interconnected services in single place. Each service have a set of function for working with images which includes such operations as creation and editing albums (changing the chain of images,

album's cover, its name or description), adding new images into your albums, also an opportunity to mark other users of the social network in your pictures via special tools. You can post comments to every image which you have access to.

Among different social services there is a group of special services called photo hostings. They are created specially for sharing images and have an extended set of functions. We will consider the most popular services among photo hostings Flickr [1] and Picasa Web Albums [8].

Picasa Web Albums provides an opportunity to combine your images into albums which could be also combined into galleries. You can add a title and a description to each album, mark some images as "private" or "public" to confine an access to them. Similarly to social networks you have an opportunity to mark other users in your pictures and comment all available images. Each picture on the service could have a set of tags which helps to find pictures by different categories. Picasa Web Albums also implements a unique mechanism of recognizing faces on photos. This function helps to create albums with definite images.

Flickr service also gives you an opportunity to share your images with other users. Photostream is a special way to present user's pictures on the service. It includes all images that you uploaded to your Flickr account sorted by the date of uploading. However, every user of this service can combine his/her images by several different ways. For example, you can unite the pictures into photosets and then make some collections from these photosets. There is also an opportunity to combine your pictures into galleries.

Flickr also supports such function as geotagging. It means that you are able to indicate geocontext of your image by marking it on a special map. You can add some photos to your favorites. There is also an opportunity to add other users as your contacts and follow all changes in their collections or galleries, post comments to available pictures, join different groups, add a list of tags to each image, etc.

After consideration of different social services, we distinguished several common functions and features and developed a concept of MySocials Gallery application for browsing images from social networks. This concept is shown in Figure 2.

The User Interface allows to browse user's albums and albums created by his/her friends on social services. Therefore, we can distinguish the following functions for our application: viewing of a list of user's friends or contacts, viewing of a list of albums of specific user, viewing of a list of pictures which are included in a specific album, viewing of a list of comments for a picture and also posting comment to specific image.

At the moment MySocials Gallery is supporting interactions with four different social services: Vkontakte, Facebook, Flickr and MyWorld from mail.ru. User has to set up one or more accounts of supported services. At the beginning MySocials Gallery downloads the list of user's contacts and the list of his/her albums from customized accounts. After the user chooses one of the albums from the list the application starts to download information about all images which are included in this album. The pictures are shown as their small copies (thumbnails). After choosing one of these images for viewing it in full screen mode MySocials Gallery starts to download it's original size copy and a list of comments to that image (if this feature is supported by the social service). The application also implements an opportunity to post comments to specific picture.

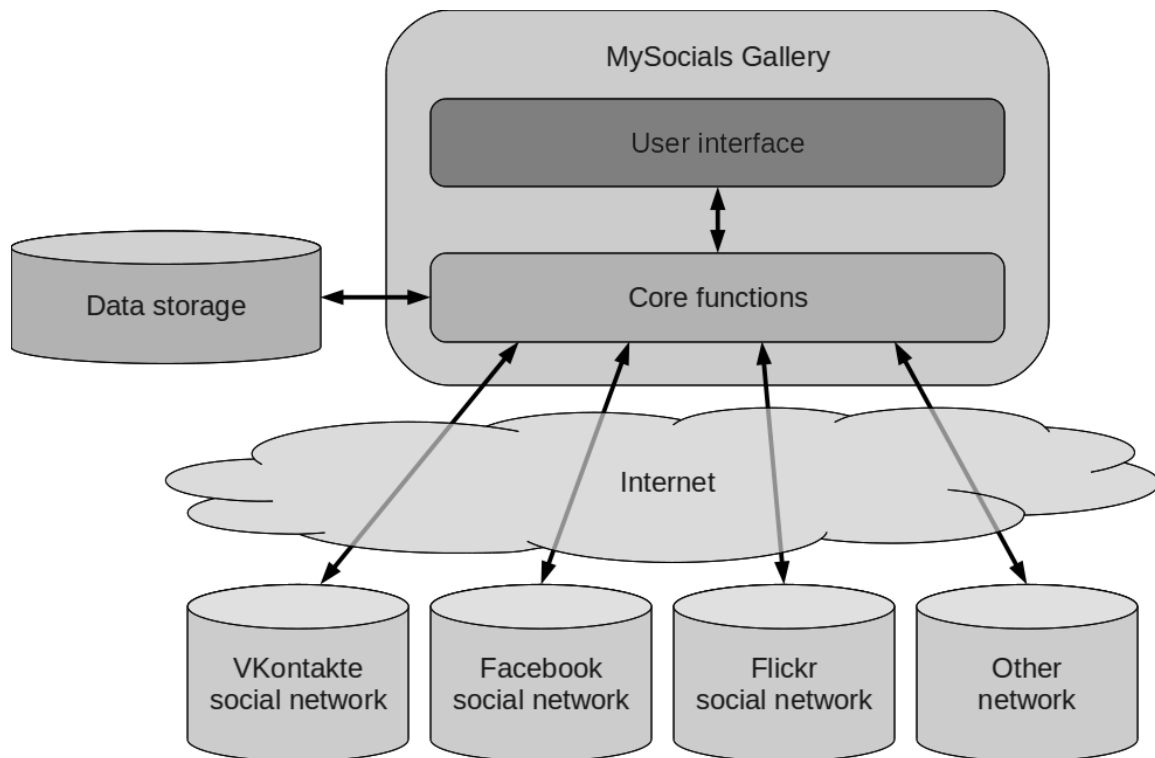


Fig. 2. Concept of MySocials Gallery application

User Interface requests all required data from Core Functions. Core Functions asks these data from one or more services using MySocials drivers. All returned service data aggregate into one response. Images which were uploaded to the social service once don't change a lot. Therefore, there is no need to download them from the service constantly. MySocials Gallery provides special caching data mechanism which helps to decrease traffic and make the work of the application faster. The application stores information about user's accounts and lists of friends into Data Storage. Also it saves information about albums and images which were browsed earlier by the user. If it is necessary the user is able to clean the Data Storage using specific button in application's settings.

V. IMPLEMENTATION

The application main form is shown on Figure 3. The form contains three panels: friend panel, album panel and image panel. One or two panels can be shown simultaneously. Friend panel contains list of friends and search string. Each item shows the profile info and avatar. Her/his own profile shows as first item in the list. The album panel contains information of selected profile (her/his own or friend) and list of albums. Each item shows album info and album image (or first image if service doesn't support album's images). The photo panel shows information of selected album and list of image thumbnails.

The image view form is opened when user select image. The form contains large size of image and list of corresponded comments. If large size of image haven't download yet then form shows image thumbnail and process message.

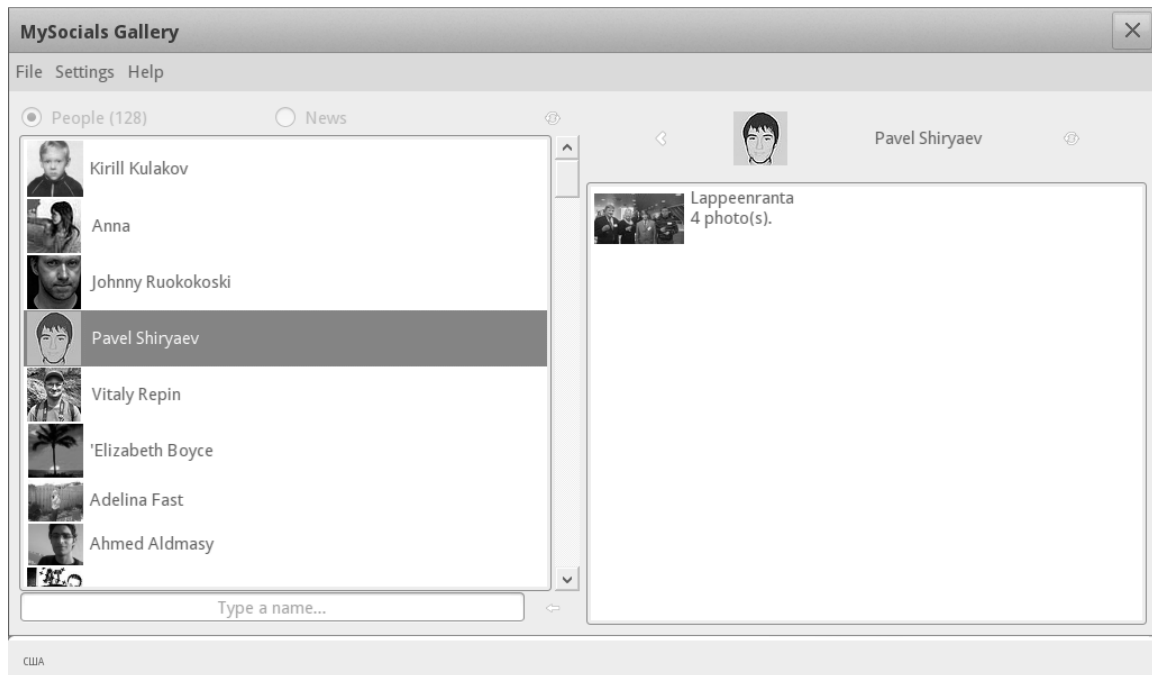


Fig. 3. Main window of MySocials Gallery

MySocials Gallery is implemented via Qt framework. Qt signals and slots are used for communication between functions. For example, when the user adds new account application generates signal "updateAccounts". One of the slots requests user's profile data. When profile data is received the signal "updateProfile" is emitted and one of corresponded slots requests lists of friends and own albums. After that the application updates all the forms sequentially.

Each request to driver is initiated in the separate thread. It allows to use User Interface without response waiting. Common list of threads is used to control access to driver and prevents the second request.

The metrics of implementation are shown in table I.

TABLE I
MYSOCIALS GALLERY IMPLEMENTATION METRICS

Project	LOC	Comments	Developers
MySocials Gallery	383 (Qt)	66	4
MySocials Core	5048 (Qt)	532	5
MySocials drivers	1589 (C)	537	3

The small size of code for MySocials Gallery is associated with the inclusion of the UI elements (forms, panels and widgets) into the MySocials Core package. The remaining part implements connections between UI elements and additional functionality (for example, help support).

Base data structure used in MySocials Gallery is a Qt class. There are classes for such elements as album, image, comment, etc. All objects are connected via special identifiers (ID) which are taken from social services. For example, instance of class *Comment* includes following sequence of ID: *Image ID* – an identifier of the image which was

commented, *Album ID* – an identifier of the album which contains this image, *Owner ID* – an identifier of the user that created this album in social service, *Account ID* – an identifier of the user’s account. That sequence is shown on Figure 4.



Fig. 4. Internal connection between classes of MySocials Gallery

One of the most important features of MySocials Gallery is an offline mode. The user is able to browse those images that he/she has looked through before even if there are some problems with network connection on user’s device. Special caching mechanism is implemented in the application. It is based on XML format. All necessary data about user’s friends, their albums and images is stored in the memory as XML files with definite structure.

VI. CONCLUSION

We described the current state of MySocials Gallery project. It supports four social services such as VKontakte, Facebook, Flickr and MyWorld from mail.ru. The full code is available at Gitorious [9], a hosting for distributed open source projects. MySocials Gallery is available for mobile platforms Maemo 5 and MeeGo and for desktop Linux based operation systems (e.g. OpenSUSE, Ubuntu, Fedora and Mandriva).

The main advantage of MySocials Gallery application is that the user can work with pictures published in different social network simultaneously. For example, when you use an Internet browser you have to open several tabs at the same time and switch between them. Therefore, using of only one application to follow all your friends is very convenient.

Nowadays there are many various applications similar to MySocials Gallery. For example, QuickFlickr [10], an application for browsing images from Flickr. MySocials Gallery has several distinguishing features. First of all, it allows to work with different social services simultaneously and combine the lists of contacts and albums into one list. Secondly, MySocials Gallery supports the work with proxy server. Thirdly, this application uses a special client WebAuth [11] to pass the authorization on the service via web interface. It means that MySocials Gallery doesn’t store user’s registration information (e.g. login and password) if it is not needed. This approach increases application safety.

There are many ways how to develop the project further. We are planning to add albums which include user’s favorite images and images in which the user is marked, because these functions are very popular in social services. Also the application will support such features as browsing news from social networks, making meta-contacts, etc.

ACKNOWLEDGMENT

Authors would like to thank Finnish-Russian University Cooperation in Telecommunications (FRUCT) program for the provided support and R&D infrastructure. We would also like to thank Sergey Balandin, Timofey Turenko and Yurii Bogoyavlenskii for providing feedback and guidance.

REFERENCES

- [1] Flickr. March. 2011. [Online]. Available: <http://www.flickr.com/>
- [2] Vkontakte. March. 2011. [Online]. Available: <http://www.vkontakte.ru/>
- [3] Facebook. March. 2011. [Online]. Available: <http://www.facebook.com/>
- [4] MyWorld@mail.ru. March. 2011. [Online]. Available: <http://my.mail.ru/>
- [5] Open Platforms for Mobile Devices. By Petrozavodsk State University, Department of Computer Science. March. 2011. [Online resource]. Available: <http://oss.fruct.org/>
- [6] Open innovations framework program FRUCT, March. 2011. [Online]. Available: <http://fruct.org/>
- [7] MySocials development wiki. March. 2011. [Online]. Available: <http://oss.fruct.org/wiki/MySocials/>
- [8] Picasa Web Albums. March. 2011. [Online]. Available: <http://picasaweb.google.com/>
- [9] MySocials page at Gitorious. March. 2011. [Online]. Available: <http://gitorious.org/mysocials/>
- [10] QuickFlickr. March. 2011. [Online]. Available: <http://www.d-pointer.com/solutions/quickflickr/>
- [11] WebAuth development wiki. March. 2011. [Online]. Available: <http://oss.fruct.org/wiki/Webauth>