ONE SOCIAL

Sowmya Sampath
Agenda

- Motivation and Project Description
- Related Work
- Design, Configuration and Tools used
- Implementation
- Demo
- Tests and Observations
- Conclusion
Motivation and Project Description

- Facebook and Twitter combined have more than 700 million users.
- Most common for users to create accounts in both Facebook and Twitter.
- **OneSocial**
  - Provides a single view for Facebook statuses and Twitter tweets.
  - Threads Facebook and Twitter feeds together
  - Updates both Facebook and Twitter with a single click.
  - Publishes an web service API that helps further integrations with other networks and helps external developers.
Agenda

- Motivation and Project Description
- Related Work
  - Design, Configuration and Tools used
  - Implementation
- Demo
- Tests and Observations
- Conclusion
Bing Social and Threadsy

Kelly and Madeline are looking forward to seeing friends this weekend.

Sergei Shevlyagin A bit too much rain but still a great ride!

Emma Laio checked out images of “north cascades national park”.

Bob Kelly is amazed at the damage a determined bear can do to a campsite.
Related work

- Bing Social
  - Collects the feeds from both Facebook and Twitter and displays them in reverse chronological order.
  - It has no facility to update in Facebook or Twitter.

- Threadsy
  - Collects the feeds from both Facebook and Twitter and displays them in reverse chronological order.
  - In addition, has a facility to update individually to Facebook and Twitter.

- Both Bing Social and Threadsy offer single sign-on facility.
## Feature Comparison

<table>
<thead>
<tr>
<th>Applications</th>
<th>Update to both Facebook and Twitter</th>
<th>Read from both Facebook and Twitter</th>
<th>Thread Facebook and Twitter messages</th>
<th>Publish the services as a REST API</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bing Social</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Threadsy</td>
<td>Yes (separately)</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Facebook to Twitter</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Twitter to Facebook</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>OneSocial</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Agenda

- Motivation and Project Description
- Related Work
- **Design, Configuration and Tools used**
- Implementation
- Demo
- Tests and Observations
- Conclusion
Design

- OneSocial provides a layer of Abstraction over both Facebook and Twitter
- Common services are placed in the base class
- OneSocial was built upon MVC design pattern
- Easy to integrate with more social networks
Configuration and Initial Settings

- OneSocial Application is created in Facebook and Twitter, which yields the secret key.
- Other settings like https protocol, creating SSL certificates, port forwarding were done to host OneSocial website securely on the internet.

Tools Used

- Zend Platform with PHP with Apache, MySQL, HTML, CSS and Javascript were used.
Agenda

- Motivation and Project Description
- Related Work
- Design, Configuration and Tools used
- **Implementation**
- Demo
- Tests and Observations
- Conclusion
Implementation – Update Statuses

- Users link their OneSocial ID with Facebook and Twitter using OAuth

- Upon successful authentication, users enter their status update in OneSocial

- Call Facebook’s Update API passing the status message and user’s Facebook access token

- Call Twitter’s Update API passing the status message and user’s Twitter access token
Implementation - Threading

- Convert Facebook and Twitter stream into a common OneSocial stream format

- Group Twitter tweets by identifying Root, Comment and Sub-Comment tweets using id and in_reply_to_status_id of each tweet

- Group Facebook and Twitter by their Root streams’ creation time

- Perform a numeric sort (reverse chronological) on the grouped feeds

- Group messages (and sort comments) that have the same text
Implementation – Web Service API

- Web page and Web service execute same code path with different output format

- APIs use the common OneSocial stream structure

- Set Updates API – Write status message to both Facebook and Twitter

```
curl -k "https://onesocial.dyndns.org/api/v1/setupdates" -d
"userid=master&pwd=*****&status=Example Status Update"
```
- Set Comments API – Set comments for a particular status message

```
curl -k "https://onesocial.dyndns.org/api/v1/setupdates" -d "userid=master&pwo=*****&statusID=100001530595145_158042127570475&comment=Response thru API"
```

- Get Updates API – Displays both Facebook and Twitter feeds in a single threaded view in reverse chronological order.

```
curl -k "https://onesocial.dyndns.org/api/v1/getupdates" -d "userid=master&pwo=*****"
```
Agenda

- Motivation and Project Description
- Related Work
- Design, Configuration and Tools used
- Implementation

**Demo**

- Tests and Observations
- Conclusion
Demo

OneSocial Website
Agenda

- Motivation and Project Description
- Related Work
- Design, Configuration and Tools used
- Implementation
- Demo

Tests and Observations

Conclusion
Tests and Observations

- **Unit Testing**
  - Tested individual functions such as getUpdates, setUpdates, getComments, setComments and Authentications using various PHP scripts
  - Passed both valid and invalid credentials to these functions and validated the behavior
  - Set various comments and sub-comments to validate the threading functionality.
  - Helped identify change in response structure from id to id_str and in_reply_to_status_id to in_reply_to_status_id_str

- **API Testing**
  - Used various test PHP scripts with CURL to validate the API responses
  - Simulated network outage and validated that the response timed out
  - Verified the APIs return 200 on success, 404 for user not found and 401 for invalid authentication.
Usability Testing

- **Test I** – Locate Facebook and Twitter statuses in Threadsy and OneSocial.
  
  **Result I** - The time taken was not much different for Threadsy and OneSocial.

- **Test II** - Update a status "I am going to hawaii this weekend" in both Facebook and Twitter using Threadsy and OneSocial.
  
  **Result II** – Threadsy took a little longer than OneSocial as users have to repeat the same message.

- **Test III** - Read all the comments posted for the status above in both Facebook and Twitter.
  
  **Result III** - In Threadsy, users had to navigate and scroll for more than a minute to collect the statuses and its comments.
Time taken for User Activities in Threadsy vs OneSocial

- Users to locate Facebook and Twitter feeds
- Users to update the same status in Facebook and Twitter
- Users to read the comments for the same message in Facebook and Twitter

Millisecs

OneSocial
Threadsy

User Activities
Agenda

- Motivation and Project Description
- Related Work
- Design, Configuration and Tools used
- Implementation
- Demo
- Tests and Observations
- Conclusion
Conclusion

- Managed multiple social networks easily
  - One Social combines and threads Facebook and Twitter stream together
  - Updates both Facebook and Twitter in a single click

- Published REST APIs to build new social networks
  - Easily scalable to include new social networks by changing its feed format to OneSocial response format
OAuth

- Redirect the user to https://graph.facebook.com/oauth/authorize and pass the application id and post authorize callback URL as parameters to this API.

- User enters their credentials for Facebook and Twitter are authorized and a verification string is returned along with the callback url.

- Generate an access token using the above verification string by fetching https://graph.facebook.com/oauth/access_token

- Once the access token is obtained, all further communications to Facebook/Twitter will only require this token instead of the user’s credentials
Twitter Response Structure

```xml
<statuses type='array'>
  <status>
    <created_at>Fri Jul 16 16:58:46 +0000 2010</created_at>
    <id>18700887835</id>
    <text>got a lovely surprise from @craftybeans</text>
    <source>web</source>
    <in_reply_to_status_id></in_reply_to_status_id>
    <favorited>false</favorited>
    <user>
      <id>29733</id>
      <name>cindy li</name>
      <screen_name>cindyli</screen_name>
      ...
    </user>
  </status>
  ...
</statuses>
```
Facebook Response Structure

```
[id] => 1437286239_172765289409397
[from] => ( 
    [name] => John Doe
    [id] => 1437286239
 )

[messages] => It is so cold in SFO this winter
[type] => status
[created_time] => 2010-11-26T19:10:59+0000
[updated_time] => 2010-11-26T22:26:20+0000

[comments] => ( 
    [data] => ( 
        [0] => Array ( 
            [id] => 1437286239_172765289409397_2081304
            [from] => Array ( 
                [name] => Will Rafael
                [id] => 3434451
            )
            [message] => Welcome to my life in New York
            [created_time] => 2010-11-26T21:11:12+0000
        )
        [1] => Array ( 
            [id] => 1437286239_172765289409397_2081655
            [from] => Array ( 
                [name] => Glen Mcgrath
                [id] => 634622076
            )
            [message] => How cold is it now?
            [created_time] => 2010-11-26T22:26:20+0000
        )
    )
    [count] => 2
)```
<message>
   <id>1437286239_172765289409397</id>
   <image>https://graph.facebook.com/1437286239/picture</image>
   <user>Kalyan Sundaram</user>
   <text>Break the Rules for next 6 months.....</text>
   <app>Facebook</app>
   <time>12 hour(s) ago</time>
   <likes>1</likes>
   <comments>
      <image>https://graph.facebook.com/1437286239/picture</image>
      <user>Kalyan Sundaram</user>
      <text>
         @Harjot : Yes
         @Bala : yes, early start to enjoy the Thanks Giveing :-)
         @Maha : Cheers .. Party lam already started. unforgettable
         start :-)
      </text>
      <app>Facebook</app>
      <time>51 minute(s) ago</time>
   </comments>
</message>