Introduction to Software Engineering
Easy Cloud

Rishabh Nigam(10598) Pankaj Prateek(10497)

Nov 18, 2013

1 Introduction

Easy Cloud serves as a replacement to the paid services provided by Cloud Storage Services such as Dropbox, Google and Skydrive. Through this system, we aim to provide the user an abstract layer over various cloud services, which would give the impression of a larger amount of data being accessible on the cloud by combining the spaces provided by these services.

Though there are various paid services which provide a larger amount of cloud space, but they charge too much. On the other hand, people have free space available at various cloud services. The initial motivation was to use this space. The idea is to make it in a more robust and scalable way, providing the user a drive like structure (like Dropbox and Google Drive desktop clients).

EasyCloud provides you a drive on your local disk which is synced to one/more of the online storage sites. The files are currently synced to Dropbox, Google Drive and Skydrive. The target audience for this project includes all the people who are using cloud storage services.

Currently, the drive has been implemented and throughly tested for linux. We plan to extend the support to windows in upcoming releases.

2 Functionality

The app on installation asks the user to create a folder named "easyCloud" and put the files it wants to sync in this folder only. There needs to be a same named folder in the cloud locations in Dropbox, Google Drive, and Skydrive. The easyCloud folder on the local drive is synced with the combination of corresponding folders on {Dropbox, Google Drive, and Skydrive}.

We have a basic GUI which allows user to choose some set of sync settings which are as follows.

1. Upload Location. The following options are available.
    - Split: It stores the file in exactly one of the place on the cloud. This is decided by an algorithm which is described in later sections.

- All: This simultaneously updates the file to all the places on the cloud which are authenticated by the user
- Only to Dropbox.
- Only to SkyDrive.
- Only to GoogleDrive.

2. Sync:
   - A sync button is provided on clicking which the files in the local drive are synced to the files on the cloud.
   - A sync is also carried out every 15 minutes.
   - Event driven sync is also implemented. In this case, if the application detects a change in the file structure of the easyCloud folder on the local drive (by comparing it to an older version of the list every 2 seconds), the files are synced. Such event driven sync can also be implemented for cloud services, but getting the file lists over the cloud takes a considerable amount of time causing the application to become inresponsive.
3. Authentication: There are 3 buttons provided in the GUI to authenticate each of Dropbox, GoogleDrive and SkyDrive repectively. On clicking those button, the user is authenticated using the stored authentication tokens. If the tokens are not found or have expired, instructions are given to the user to authenticate. The user is provided with a authentication link, which needs to be opened in the browser and the code returned in this process, works as an authentication token for the user.

3 Implementation

The complete implementation is done in python 2.7. *python-tkinter* is used for GUI. The API's of various services are taken from their developer websites.

1. A folder named "easyCloud" is created in the home folder of the user which is synced with folder of same name in the home directories of each of the linked accounts.
2. If a file is not present in any of the cloud services, but is found in the local folder, it is uploaded to the cloud. The location is decided by the location algorithm.
3. If a file is present in some cloud service but was not present in the local folder previously, it is downloaded to the local folder. This would happen when a user directly uploads the file in the cloud service (using web/other services).
4. But if a file is present in some cloud and was also present in the local folder previously, it is deleted. This would happen when a user deletes a synced file from the "easyCloud" folder.
5. **Location Algorithm**: The user is presented with 5 choices, "Split", "All", "Only To Dropbox", "Only To skyDrive" and "Only To Google Drive". When any of the last four is selected, the file is uploaded accordingly to the selected choice. But if "Split" option is selected, the application checks which of the linked accounts has the highest (free space/total space) ratio and the file is uploaded in the corresponding account.
6. **Sync**: There are two sync options, in the first, the folders will automatically sync every 15 minutes, and the other is event driven, i.e., when a user uplaods or deletes a files from the local folder, the folders are synced. Apart from this, the user is also presented with a button in the GUI which would sync the folders.

4 Testing

The application has been extensively tested on various linux machines. Various combination of tasks like upload, download, delete using various sync settings have been tested. Some bugs and inconsistencies were found in the testing processs. These were corrected followed by a subsequent phase of testing.

5 Furthur Work

1. Port the application to Windows and other operating systems.
2. Include other services like box.net, copy.com, amazon cloud etc.
3. Develop and implemenet better algorithms for dividing files over various cloud services.

6 References

- Dropbox: https://www.dropbox.com/developers
- Google Drive: https://developers.google.com/drive/
- SkyDrive: http://msdn.microsoft.com/en-us/library/live/hh826521.aspx
- The project had been hosted on Github at https://github.com/pankajprateek/easyCloud