Deign Document: Basic Layer – File I/O

Hang Yuan (hyuan211@gmail.com)

Version: 0.1 (08/05/19)

1 Goals

The project will be based on our own designed database to store the email message instead of other existing standard libraries. Just like the other modules in each layer of this project, the "File I/O" will work as a functional module or library to the down layer (Manager Layer).

The File I/O module will provide other modules accesses to the file contained locally where store all the information of the system.

2 Design

The design for File I/O module will be included in three parts: (1) system design; (2) file type and format; (3) unit tests plan.

2.1 System Design

The module uses <cstdio> to open, create or delete file. This unit doesn't deal with the actually reading or writing.

Basic functions will be:

```
createFile(const string &filename)
destoryFile (const string &fileName)
openFile (const string &fileName)
closeFile(FILE *_fd)
sysfile_auto_checker()
```

Beyond the basic operation to file accesses, this module will also auto-check the file list under the 'data' directory to initialize files as the initialization to system need. If the auto file-checker status reports error (for instance, access deny or lacking necessary files), the system will handle any issue properly, either fix it or throw an error. The data directory must include following files, if no such file, the system will automatically create a new one in the relevant file type:

Client (user end) side	Server (service provider) side
email.data user.sys config.sys log.sys	<username>.data user.data log.sys

More information about the type of file will be covered in next section.

Based on the ongoing development and project functionality requirements, more files might be added.

2.2 File Type and Format

This project uses the self-designed database to control the data collection, modification, and reading access. So we design the specific type of data and corresponding format to support the data accesses.

2.2.1 File types

There are only two types maintained in the data directory: .data and .sys.

.data: the file contains the project data, including the users' information and the email content (with email header) for each account. All data will be stored in plain text (we will consider adopt cipher text to improve the security in the future).

.sys: will contains the system file, which cannot be modified at most time. This type of file only records the system configuration and actions for other modules to use.

2.2.2 File format

All files, no matter what the type, will contain a file header to store the basic file information. Different file with different purposes will have different usage which will be covered in other design documents.

2.3 Unit Test Plan

Unit tests will be run on the UNIX environment.

Unit tests for File I/O will cover nearly all possible cases that the module will meet. The unit test plan will include two parts:

(1) Valid and invalid input

(2) Edge cases:

a) Access issue

b) Lacking necessary file(s)

c) Creating file with existing filename

d) Opening nonexistent file

e) Reading nonexistent file

f) Closing nonexistent file

g) Deleting nonexistent file

All unit tests have been included in bl/basicIO.cpp.

A Makefile is provided to generate test result log.