Project Proposal

A Desktop Peer to Peer Photo Sharing Application

Ryan Adams
CSE 403
Winter 2007

Operational Concepts

This project is to create a photo sharing application. The application would run on a desktop machine and would act as a client and server. It would communicate with other computers running the same application using TCP/IP. The program itself would be able to organize, view, and share pictures that were stored on the local computer's hard drive. It would also be able to connect to other people's computers that are running the same application and view their pictures (as long as this was allowed by the other person).

The program would have to have privacy controls and it would need to work well on slow and fast networks. This means that a person who is browsing someone else's pictures should only receive small thumbnails of the pictures at first and then be given the option to download the picture(s) in different resolutions.

This project is aimed at people who:
- have a large collection of pictures that they want family members and friends to see
- are on a reasonably fast and always on network (LAN) or internet connection (WAN)

This software would have an additional appeal to people who:
- don't want to pay for an online photo sharing program
- don't want to upload their photos to a online service or other server
- want to share their pictures over a LAN at high speeds

System Requirements

The program would have the following features at the basic level:
- A Graphical User Interface that allows the following to be displayed:
    - Albums as thumbnails or as a list
    - Pictures within an album as thumbnails or as a list
    - A picture by itself with zoom and pan functionality
    - Other Computers
- The ability to add, manage, and remove albums where an album is a collection of pictures that already exist on the hard drive.
- The ability to connect to another computer by entering a host name or IP address
- The ability to view albums and pictures on other computers (once connected) and the ability to save those pictures to local files.
- The ability to act as a server and process requests from other computers (users) such as:
    - A request for a list of albums
    - A request for a list of pictures within an album
    - A request for a picture at a certain size (in this case the server would resize the picture on the fly and then transfer it)

- The ability to add, manage, and remove users.
- The ability to restrict the sharing of photos and albums on a per user basis.

These features could be added if there was time:
- The ability to manage, store, and share extra meta data about each image
- A secure connection between client and server
- The ability to search for pictures

Here is an example network situation between 4 computers running this software. User A is connected to User B and has requested an album list. User C is connected to A and D. User C is requesting a picture from A and is receiving a picture from D. User D is connected to User C and has requested a picture list, which is now being received:
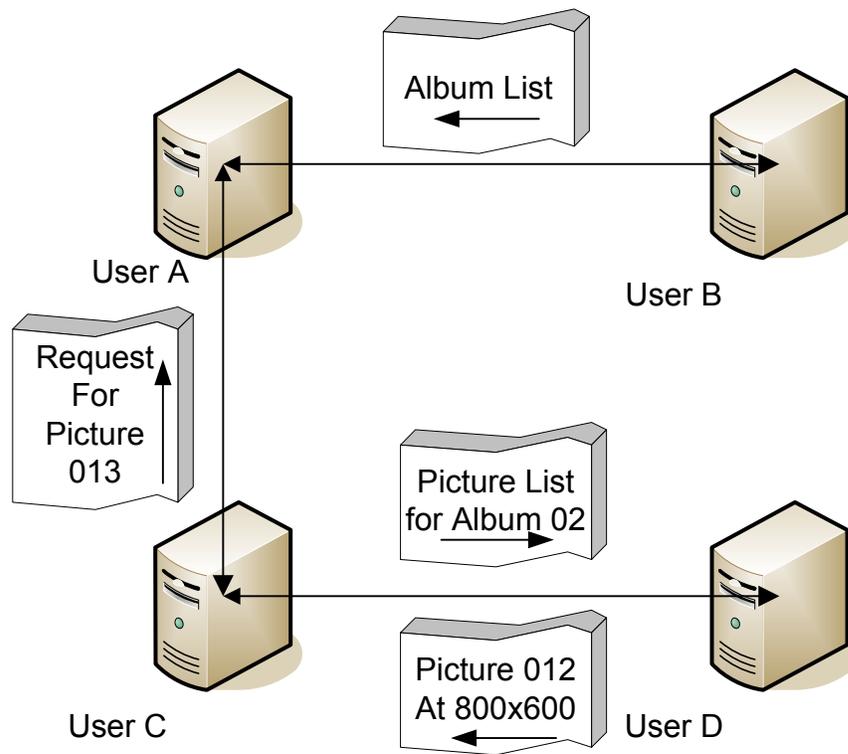
System and Software Architecture

The current plan is to implement this project in JAVA because it has good IDE tools that would help the overall software development and because JAVA is sufficient to accomplish all of the requirements. The plan would be a stand alone JAVA application. There would be no other software parts, everything would be contained in the JAVA application.

Lifecycle Plan

The major goals for this project can be expressed as components of the program that should be implemented:
- A working client that can view and manage its own pictures and albums
- The client component that manages users and permissions
- A graphical library that can resize pictures
- A server component that can provide pictures and lists in accordance with permissions
- The client component that can connect to servers and browse albums and download pictures

These tasks should be completed in order but many can be worked on at the same time. The project will likely require 4-6 workers in order to complete it in 8 weeks. The following skill sets will be needed:
- Experience with network protocols (specifically implementing or using one in java)
- Experience with images and image manipulation (manipulation algorithms, file formats, compression)
- Experience with Java GUI components
- Experience with Java I/O libraries
- Experience with image format java libraries (i.e. gif, jpeg, png, bmp, tiff)

Feasibility Rationale

Java and associated libraries should provide all of the tools necessary to complete the project. I don't think any of the requirements are impossible to do. However, we could easily run into time problems. It is hard to say whether the project will fit in the time frame without working through a complete specification.

If the project were to become behind schedule at any point, some of the complexity of the project could be removed. Specifically, the requirement of having a user / password permission system could be dropped. This would make the project more of a proof of concept because no one would use it if they could not protect their images. However, it would still be a functional program that could demonstrate the main goal of sharing pictures.