# A Metadata-orientated Integrated Approach to Personal File Management

***Abstract-*** **In recent years, as the computing and storage technology developed, personal information is now scattered over different systems and devices in different forms and formats. We take help of the file metadata to formulate a solution to personal file management. The currently-used file synchronization and backup requires active involvement of a user. Our method relies on a light-weight client program which passively identifies a user activity during any login session such as file creation, edit, and deletion (with the relevant file metadata). The client synchronizes such information with a metadata repository and optionally, a centralized file-storage. Our approach can easily be extended to support multiple users. Our basic idea of metadata orientated personal file management with an implementation that can handle some crucial personal file management problems like version tracking and duplication detection, *et al.* We also provide with the underlying algorithms. We also elaborate our future plans to address more complex file management scenarios using the same framework.**

## I. Introduction

Personal Information Management is a multidisciplinary research area. Personal information items include things such as personal files, e-mails, bookmarks, calendar entries, contacts, etc. The aim of PIM is to reflect the model in storing and retrieving the personal information. We address personal file management problems that arise from the fact that people often use more than one computer, and so personal files are generally scattered across the systems they use. There is no proper way to know what files a user owns and where they are located at. Users generally have multiple copies of their files across different systems or different folders of the same system. There is an urgent need to organize all users' files at one place in a systematic fashion with a feature provided for intuitive search and navigation. Users generally store information in files which are subsequently stored and managed with the help of an operating system. From the user point of view, files are often organized in a folder hierarchy of the file system. File related metadata offers rich information about file. Such metadata can be efficiently used to help users organize and locate all their files at one place without much of effort. We explain client-server architecture for personal file management that requires minimum user involvement and is able to organize all the files of a user at one place over time. We develop a light-weight client program which authenticates a user at the login and runs in the background. The client records all the file activities of a user. At certain interval or during the logout process, the client aggregates and synchronizes all the file events with the server along with relevant metadata in XML format. A metadata repository which can also be configured is generally known as the server.

The client program must be installed on the system. The approach explained here handle multiple users. In the following section of this paper, we review research background. We explain the file metadata used in our implementation in Section III; and the software development itself is explained in Section IV. Section V analyzes the results in personal file management. Finally we conclude the paper in Section VI.

## II. Research Background

### A. MyLifeBits

MyLifeBits is a project to fulfill the Memex vision. It is a system for storing all of one's digital media, including documents, images, sounds, and videos. It is built on four principles: (1) collections and search must replace hierarchy for organization (2) many visualizations should be supported (3) annotations are critical to non-text media and must be made easy, and (4) authoring should be via transclusion. The MyLifeBits project is an effort to implement a personal digital store. MyLifeBits is a database of resources (media) and links. A link indicates that one resource annotates another. MyLifeBits uses SQL Server with Index Server supplying full-text search. Text searches can be performed over resource descriptions. After storing all the content and metadata from various types of application such as contacts, documents, email, events, photos, songs, and video in the SQL database, it tries to organize the items, all captured entities exposed to the user, in a single huge folder rather than categorizing in many folders and sub-folders.

### B. ROMA metadata service

ROMA is a service which allows people to switch among multiple heterogeneous devices and access their personal files without dealing with nitty-gritty file management details such as tracking file versions across devices. This goal is achieved through the use of a centralized metadata repository that contains information about all the user's files, whether they are stored on devices that the user himself manages, on remote servers administered by a third party, or on passive storage media like compact discs. Metadata is stored in XML format, and it uses XSet, a high performance, lightweight XML database, for query processing and persistence. There are three Roma-aware applications currently built.

### C. MDMS

MDMS provides a data management and manipulation facility for use by large-scale scientific applications. Preliminary results obtained show negligible overhead of database access time.

## III. File Metadata and ontology

### A. Basic and Application Specific Metadata Extraction

As we focus on file level abstraction in this project, file metadata is mainly used in our system. Basically file name, file size, file hash, file type and file path are included. We extract both the basic metadata, and the application-specific metadata (e.g., image height, image width, etc. for an image, video metadata, audio metadata etc). Application-specific metadata are used in effective retrieval of files through association or other relationship as defined in the file ontology.

### B. File Hash Calculation

Hashing is done on the full name of the file and the content of the file. We're using an SHA - 256 hash function.

### C. Metadata Encoding and Repository

File Metadata is stored in XML format. XML metadata repository is used to handle the metadata request and response generated from the query language MySQL in the local system.

### D. File Storage

When file objects from the client computers are stored at the server, dynamic file classification is possible with the help of file metadata. File organization using folder and sub-folder structure at the server side is therefore not necessary. Server stores all files of a user in a flat list of file objects with unique fileID. In this way, both the system and the user can find all of a user's files at one place regardless of how many systems and devices they use.

## IV. Software Development

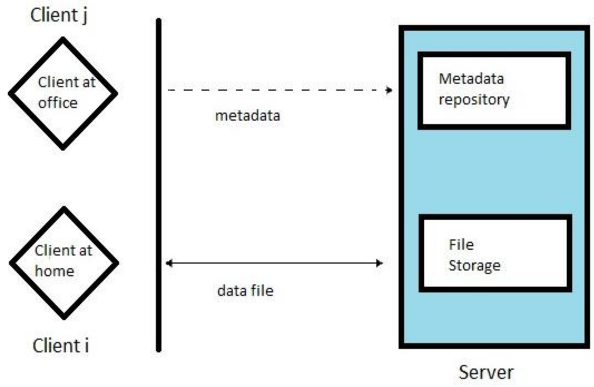
To implement the concept of our work, we focus on the metadata processing.

### A. Working Methodology

Metadata of the files (general and application) are extracted by using the header files <windows.h>, <objidl.h>, <jdiplus.h>, <stdafx.h>. Queries were written in SQL and the query was invoked through the interface and stored in database. Hashing is done on the file ID and the file contents. A SHA – 256 hash functions was used. There are no known collisions in the SHA-2 family of hash functions till date. File structure is made up of folders, sub folders, files and so is implemented using graph. Files are divided into different categories media, image, document, others. Media is further divided into video and audio. Folder's name is taken as the input and metadata of all the files are stored in the file general table and is recursively done for all the sub folders. In this process all the file's specific metadata are categorized according to file type and stored in the respective tables. Breadth first search algorithm is implemented to search for a file user asks for, in a directory. If the file user wants, is not present in a particular directory, files of that directory are removed from the memory and the subfolders (nodes) of that directory are stored in the memory one by one. Based on user's action (delete, update, or create), the metadata will be updated in the repository.

### B. Integrated PIM Architecture

The client server model which explains our proposed architecture is given below:



### C. Research Gap:

Paper describes PIM for single User. The group implements the same for multiple users. Systems were connected with WiFi. One System hosts as a server. The other systems can access the database by entering the IP v4 address of the server in the browser and then by logging into myPhp admin will have specified privileges.

## V. Results

The software is able to achieve the required purpose of file management. The metadata is extracted from the files and stored in database and the according to users' needs a new file can be created, a file can be deleted or a file can be updated with appropriate changes visible in the database too. The idea was extended to multiple users with successful results as mentioned in the above section.

## VI. Conclusions

The method is not merely the file synchronizer or backup, it is trying to improve the present solutions for personal file management problems based on the metadata. As a future work, one can try to cover all kinds of personal information in PI (Personal Information) space. Also file classification based on file type can be done. Saving the same file with a different file name will not carry the version so they are not making reference to the same file which is an issue to be looked upon. Lastly utilizing the metadata kept in server part can provide users with vast information about their files.

## VII. References

- Md Maruf Hasan, Chutiporn Anutariya, M Zau Ja; A Metadata-orientated Integrated Approach to Personal File Management; School of Information Technology Shinawatra University Bangkok, Thailand
- Gemmell, G. Bell, and R. Lueder, "MylifeBits, a personal database for everything," CACM publication, Microsoft Bay Research Center, Sanfransico, CA, January 2006
- Wei-keng Liao, Xaiohui Shen, and Alok Choudhary; Meta-Data Management System for High-Performance Large-Scale Scientific Data Access; Int. Conf. on High-Performance Computing 2000
- Edward Swierk, Emre Kıcıman, Vince Laviano and Mary Baker; The Roma Personal Metadata Service; Stanford University Computer Science Department Stanford, CA 94305 USA

## VIII. Group 10

1. Snehil Shwetabh 2011C6PS502P
2. K. Sachin 2011C6PS661P
3. Arvind K. 2011C6PS666p
4. Akhil Tripathi 2011C6PS739P
5. Nitin Suri 2011C6PS802P
6. Rahul Priyadarshi 2011C6Ps813P
7. Harshit Gupta 2011C6PS837P