VARIATION BASED DATA MIRRORING to MIGRATION TOOL

Asra Khanam[1], Baria Shakeel[2], Khalid Makhdoomi[3]

[1]Department of CSE, SSM College of Engg and Tech, Baramulla, India

[2]Department of CSE, SSM College of Engg and Tech, Baramulla, India

[3]Department of CSE, SSM College of Engg and Tech, Baramulla, India

[1] asrakhanam12@gmail.com; [2] bariashah37@gmail.com; [3] makhdoomikhalid@gmail.com

ABSTRACT --- The problem lies in arranging the files so as to use them for retrieving or fetching data for presenting the same as information. These things are more time consuming as while managing and storing files a lot of redundancies and delicacies' happen which leads to improper data fetching later. Currently companies hire employees to collect data files, the collection of data requires more manpower as its hard to manage and organize files when they are in abundance. This process of managing files takes lot of time and manpower and still remains hard to get much of accuracy. To overcome this, we manage files in single repository and in some sort of a hierarchy so that searching becomes faster without occurrence of a hang or freeze in the system. Also backup is much faster as it creates a mirror image and fingerprints of the storage drive to identify them and to help us backup only that data which is new in the source disk. It copies files and folders without any compression, thus increasing speed. Due to this, it becomes easy to manage a lot of data in any organization.

Keywords --- Repository, scanning, storage device fingerprints, data mirroring, data migration, fast searching.

I. INTRODUCTION

This application is used to manage large and complex data repositories that traditional data processing applications are inadequate to manage. It mirrors the external storage devices and also drives of system into database uniquely and migrates the data present in the storage devices to a database, if the external storage device has been connected to the system even once, then next time the system will show all the variations in device and thus reducing the time of manual checks. It also makes searching of data more easier because initially at the time of scanning we come to know whether file is present in the storage device or not.

II.   LITERATURE SURVEY

The recent years have seen enormous progress in the field of data management and retrieval, most notably in the protocols and standards for data archival. Oak Ridge National Laboratory Distributed Data Archive Center for biogeochemical dynamics has made remarkable progress for developing modern toolsets to improve these processes.

It is often costly and tedious to migrate systems, especially for large organizations, hence they make efforts to minimize cost as much as possible. These systems vary greatly from modern ERP systems as the data access, business logic and presentation lie at the same tier.

Software engineering includes a basic methodology, that is, reverse engineering. When we monitor source relational database system, reverse engineering is the first step. In this process, we code database, document and analyze behavior to recognize the components and their dependencies for migration. The setback in this process is that there is no means to modify the system.

III.   PROPOSED SYSTEM

In our system we used SQLITE which is sever-less, zero configuration database that does not require any setup procedure. Java(J2SE 1.2) is used as it is standard edition and most widely used platform for deployment of portable code and JDBC-ODBC bridge is used to connect to the database. This application developed is a window based standalone application which organizes the complex and large data. By means of this application system uniquely identifies each external storage device or its own drives. The user inserts the USB or any Hard Disk in the system. After inserting the USB(say),the system checks for it's fingerprint i.e. whether or not the USB has been inserted before. If not, then the system creates one for this USB. System will provide a unique ID to this USB and asks for a name from the user. Thus the user remembers the USB by it's name and internally system uniquely identifies it by the ID provided.

After this, mirror of the USB will be checked in the Database. If not present, then USB is mirrored and its mirror is stored in the Database. Then the files, folders or directories present in the USB will be scanned and outputs all the files either in tree view or list view present in the USB to the users. Then the scanned files will be mirrored to the Database and thus the backup of the files will be created in the Database. Since the files are not compressed so back-up is fast. The two Databases are connected to each other by means of the mirror ID given to the USB. After it's completion, the user can search the files and view it's location and goes to that file location directly, thus saving time.

If another time same USB is being inserted in the system then the system will identify it and mirror code of USB will already be present in the system, thus the files in the USB will be scanned. Here the system will check the variations ,if any, by comparing this to the previously backed-up data present in the Database. The variations can be in the file name, file size, file path, creation date, modification date, file content, extension etc. Apart from this, some scanned files may be present in Database but not in the USB now, that means files have been deleted from the USB (on comparing, this can be shown by some color variations). If some files are present in the USB but not in the previously scanned data then the files have been added to the USB.

Based upon these variations mirroring of the files will be done like whether to mirror all files in USB again or to mirror only the files with variation or to mirror the files which have been added to USB etc. After these files are backed-up in DB, we can do the rest of the things like searching. If some file has been deleted from the USB, instead of searching that file manually one by one, we initially get to know that the file is not present on the disk anymore by means of variation in the file. Thus saving our time.

This application also creates zip of the files or directories without using a separate tool for it. Here we have to select the drive where to store the zipped file. In addition to this we can also extract the file back. Here also location is choose to extract the file. We can also transfer the zipped files to different locations.

IV.   CONCLUSION

This application manages large and complex data repositories. This standalone application makes searching and backup of data much faster. It copies data from the external storage devices such as Hard disk, Flash drives, external storage devices etc. and manages it in some sort of a hierarchy so that searching becomes faster without occurrence of a hang or freeze in the system. Also backup is much faster as it creates a mirror imaged fingerprints of the storage drive to identify them and to help us backup only that data which is new in the source disk.

ECONOMIC FEASIBILITY --- The application will surely reduce the wastage time of human resource which can be utilized for other beneficial tasks rather than manually checking the data variations on daily basis in various data repositories. In units that are managing large amount of data, it becomes a time consuming process for managers and their teams to handle and manage fast changing data, these variations in data and accessing the desired documents and files becomes a cumbersome process, thus with our application we enable them to reduce their efforts of such tasks. This way surely the Human Resource can be utilized in better tasks thus increasing the economy of the business unit.

OPERATIONAL FEASIBILITY --- In large data repositories finding proper files or documents at desired time becomes a tricky task which at times hampers the normal operations to be performed as desired, thus making searching easy and fast surely will enable companies to run their operations work or tasks on time and without any hampers due to delays in finding documents.

TECHNICAL FEASIBILITY --- In business units with large amount of data to be transferred or backed up (like data entry companies) the manual process makes the task bit complex as files and variation in data is bit hard to detect without any Technological Intervene or Involvement. Our application enable the users to find any variation in data in much quicker span of time and in well organized and human friendly ways.

REFERENCES

[1] www.google.com

[2] www.wikipedia.com

[3] www.oracle.com