Introduction

The advent of distributed file systems and the increasing computing power of portable computing devices in forms of tablets, smart phones and laptop has since changed the work place environment from the traditional desktop computers stationed in office building or the home office. In today's world a lot more work is done on the move away from office building through the use of interconnected portable computing devices. The portable devices mostly have ad-hoc or on demand connectivity to the internet and the applications used on these devices are data-centric. This created a number of unique challenges to this new computing paradigm of distributed file systems these include; the files synchronization problem, file storage the fact that portable devices have limited storage capacity and can therefore can only store limited files depending on their storage capacities as compared to their desktop counterparts, getting file access and maintaining files across the numerous devices platforms and how to deal with the file updating, conflicts and versioning.

This project implements a file synchronizer application designed to address the aforementioned challenges. The application architecture is designed to provide the most efficient methods of providing the required functionality and support different computing platforms. The architecture design decision points included; cross platform development framework, file synchronization method, file sizes and types and endpoint device capabilities. The implemented application provides file sharing of different file sizes and types, file access across different devices such as desktop computers and mobile computing devices, file updating, versioning and conflict handling, file storage both remotely (in the cloud) and locally on device storage, and lastly multiuser functionality. The application implementation in the interest of delivering the solution on time focused on achieving the core functionalities first and a few other desired functionalities that enhance the application usability.

Review and Citing

The File synchronization is a critical component to the current distributed file systems computing and is therefore a widely researched field with numerous products both commercial and open source solutions with fully embedded feature sets. A comparison study of file synchronization to determine the best file synchronization methods to solve a file transferring problem between student tablets devices and education institutional hubs provides a comparative comparison of the different file synchronizations tools Rsync, DropBox, HadoopRsync, Syncany and Unison [1]. According to this study HadoopRsync an open source framework is most suitable for meeting the aforementioned requirement. First because it can be scaled to a very large network of users and the HDFS framework used for storing data across numerous nodes provides resilience to data node failures. Secondly HadoopRsync uses Rsync [8] which provides the best file synchronization method because it handles file changes with minimal transfer overhead through tracking file and block lists both locally and remotely, transferring only the changes and reconstructing the file at the destinations. This technique was adopted and implemented in this project's file synchronizer. HadoopRsync uses peer-to-peer

file synchronization technique, client-to-server technique is used for this project because its implemented over http to provide reliable delivery and flow control.

In today's world, "the number of user managed devices continues to increase and the need for synchronizing multiple file hierarchies distributed over devices with ad hoc connectivity is becoming a significant important issue." [3] This paper reviews cloud file synchronizer architectures such as Google Drive, Dropbox, SkyDrive, iCloud, ReaiSync, Box identifies issues relating to availability, security, privacy and trust and proposes a product Cloud Enabled File Synchronizer that addresses the identified shortcomings in a single solution. This project design architecture is similar to cloud file synchronizer architectures reviewed in this work.

The file synchronization problem is a well-known challenge faced in maintaining huge chunks of files across distributed environments. Rsync [3][1] has been proposed as the suitable method and implemented widely in applications such as personal user files, business files, remote backups, mirroaring of large webpages, ftp sites and content distributions. This paper [3] however, proposes improvements to Rsync in solving the synchronization problem when dealing with very large chunks of files synchronization over slow connections.

[1] G Shiala, S K Majhib and D B Phatakc, "A Comparison Study of File Synchronization", In Proc. International Conference on Intelligent Computing, Communication & Convergence (ICCC-2015) Odisha, India.

[2] T. Suel, P. Noel, and D. Trendafilov, "Improved file synchronization techniques for maintaining large replicated collections over slow networks". In Data Engineering, 2004. Proceedings. 20th International Conference on, pages 153–164. IEEE, 2004.

[3] Aaysha Shaikh, Sonali Shetye, Kirti Pagare, & Prof. Vijaya Sagvekar "A paper Study on Cloud Enabled File Synchroniser", In Proc. Global Journal of Engineering, Design & Technology, April 2014 ISSN:2319-7293.

author   = { G Shiala and S K Majhib and D B Phatakc},
journal   = { International Conference on Intelligent Computing },
title   = { A Comparison Study of File Synchronization },
year   = 2015,
pages   = {153–164},
keywords   = {cite}

author ={ Aaysha Shaikh and Sonali Shetye and Kirti Pagare and Prof. Vijaya Sagvekar }
booktitle={ Global Journal of Engineering, Design & Technology, },
title={ A paper Study on Cloud Enabled File Synchroniser},
year={2004},
pages={153-164},
keywords={cite },
ISSN={1063-6382},
month={April},}

author={T. {Suel} and P. {Noel} and D. {Trendafilov}},
booktitle={Proceedings. 20th International Conference on Data Engineering},
title={Improved file synchronization techniques for maintaining large replicated collections over slow networks},
year={2014},
keywords={cite },
ISSN={2319-7293},
month={April},}