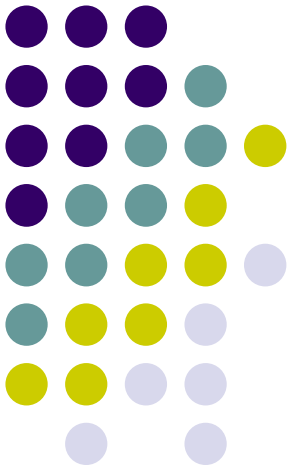
Ubiquitous and Mobile Computing CS 525M: Virtually Unifying Personal Storage for Fast and Pervasive Data Accesses

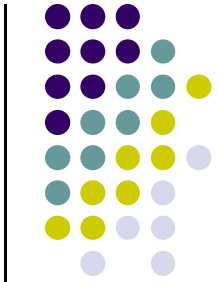
Pengfei Tang

Computer Science Dept.

Worcester Polytechnic Institute (WPI)

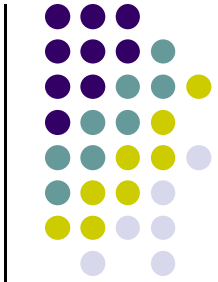


Introduction:



- With the ever-increasing processing power and ever-decreasing prices, The prediction is that there will be around 982 million smartphones in 2015
- Mobile video traffic is now over 50% of the mobile data traffic on the Internet, and it is predicted to occupy 2/3 by 2015 according to Cisco

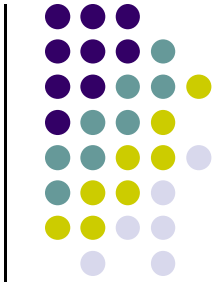
Introduction/Motivation:



- Always access data across different storage space?
  - Dropbox, Google Drive, Amazon s3, Windows SkyDrive and SME Storage, which rely on cloud-based services or a server-based approach can solve this problem in a short time

Motivation:

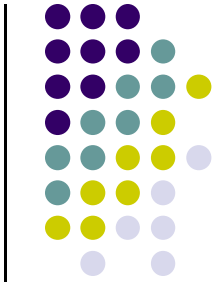
Do you feel safe about online data storage?



- Mark Zuckerberg's pictures leak incident in Facebook
- DropBox account breach with wrong passwords
- Amazon's data center failure in 2011

Motivation:

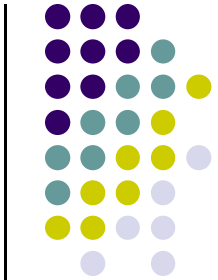
Use some other file system to avoid data risk?



- Some modern file systems have taken into account user's storage to avoid third party storage compromise. But they maintain a strong consistency model for different types of files, resulting in unnecessary and heavy performance overhead

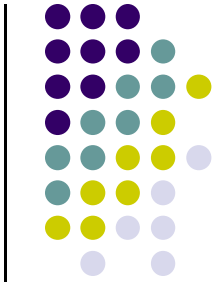
- Hence, this study aims to design and implement a system to virtually Unify Personal Storage (vUPS) for fast and pervasive file accesses

Related work



- Large scale distributed file systems, such as Goolge FileSystem (GFS) and the Hadoop Distributed File System (HDFS) have also been well studied for distributed computing
- More recently, modern distributed file systems ZZFS , Eyo, and BlueFS consider the modern portable storage devices such as tablet and smartphones.
  - ZZFS has specific data placement polices and the consistency policy to avoid conflicts, while Eyo does not guarantee any consistency.

Contributions



- vUPS provides an alternative solution to existing cloud-based or other centralized approaches for responding to the demand surge of quick and pervasive file accesses across multiple devices owned by a user
- By differentiating and treating different types of files, vUPS strives to achieve a balance between the file consistency and the maintenance overhead
- With a web browser interface and a standard file access interface, vUPS can be adopted by other applications to transparently and seamlessly access personal files

vUPS Design

- vUPS architecture
  - vUPS adapts a flexible P2P architecture

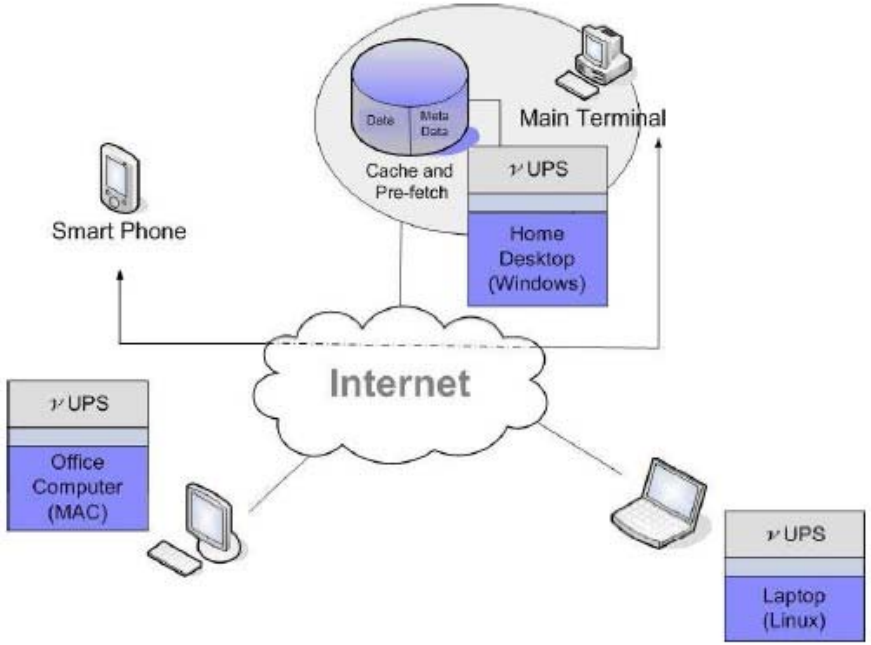


Fig. 1: The Architecture of vUPS

vUPS Design

- vUPS components

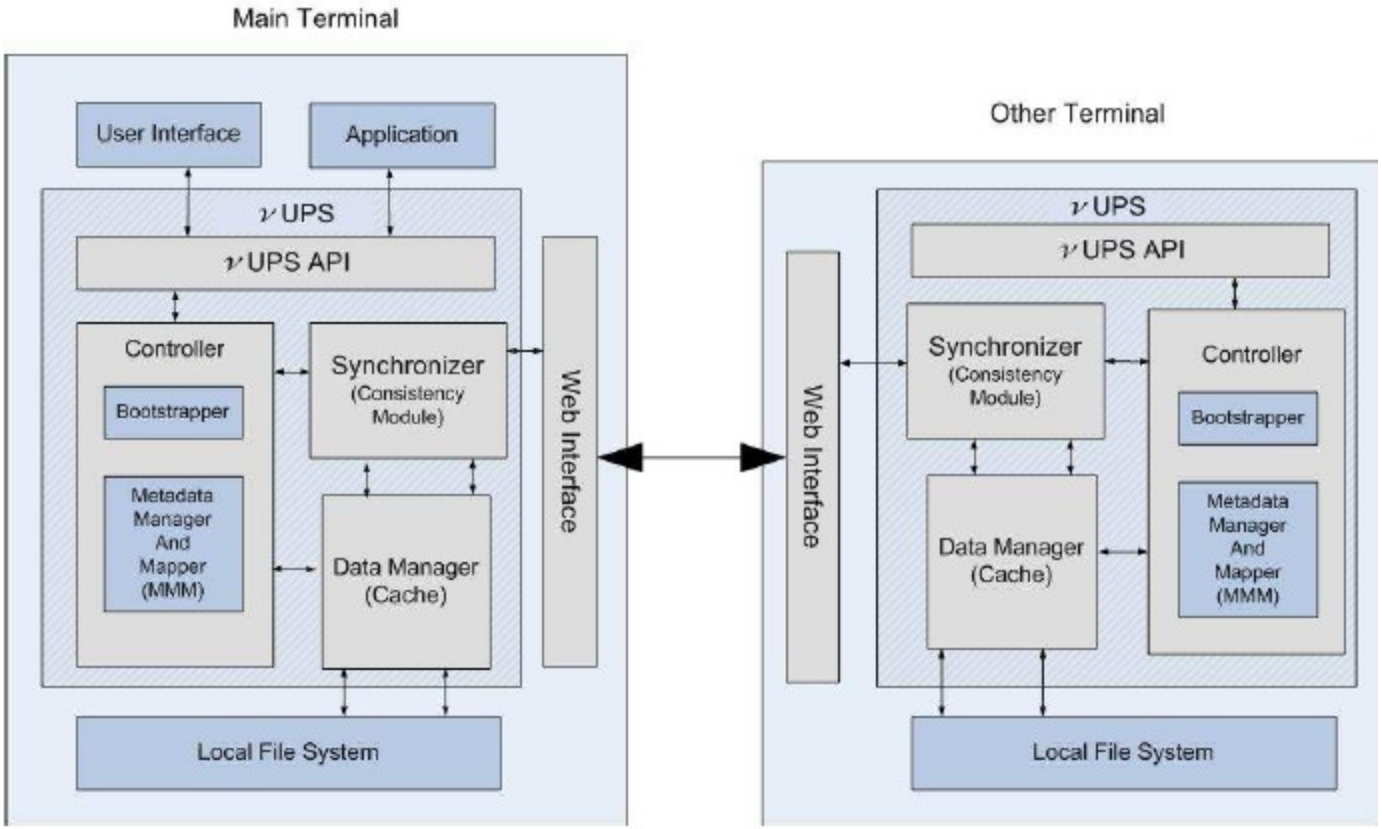
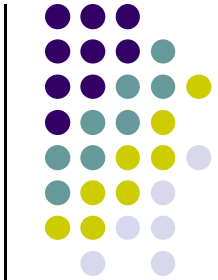


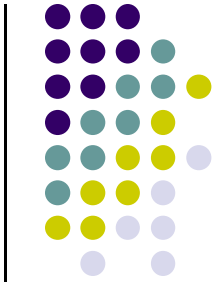
Fig. 2: vUPS Components

Consistency and Availability



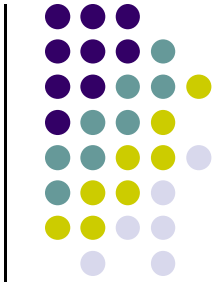
- Traditional file systems replicate the data across multiple devices to increase availability, which also increases the overhead for consistency
- Popular file systems have different policies to make the data consistent among replicas
  - Coda apply optimistic concurrency control between the replicas. Coda resolves consistency conflict using version controlling.

Consistency and Availability



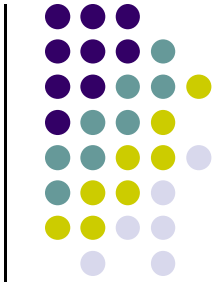
- In vUPS, files are divided into two categories: editable and non-editable.
  - The audio, video, image, and program files are considered as non-editable.
  - All other files, including doc, xml, and text files, etc., are categorized as editable files.

Consistency and Availability



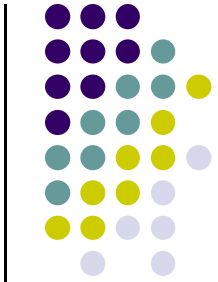
- media files are not frequently changed. Only the corresponding metadata (favorite, ratings, etc.) may be modified.
- Therefore, we may relax the consistency model for media files as they are often non-editable, while the replicas of other documents have to be consistent as they are frequently edited.

High availability with weak consistency



- Whenever a non-editable file is invoked by a device, the file is cached on the device and in the main terminal according to the caching policy. As the access-to-update ratio is higher, leaving a copy behind will improve the access performance.
- So, when that file is closed, it is synchronized (if necessary), and a copy is left in the cache. The namespace contains a callback reference to that address for providing response from that copy to other devices.

Limited availability with strong consistency



- When that file is closed, it is synchronized (if necessary), and the local cached copy is deleted. The callback reference is also deleted from the namespace once the file is closed.
- That is, vUPS enforces a strong consistency policy between the copies. It sends the update operation where data should change (active replication).

vUPS Implementation

- Implement a prototype of vUPS with Java and HTML

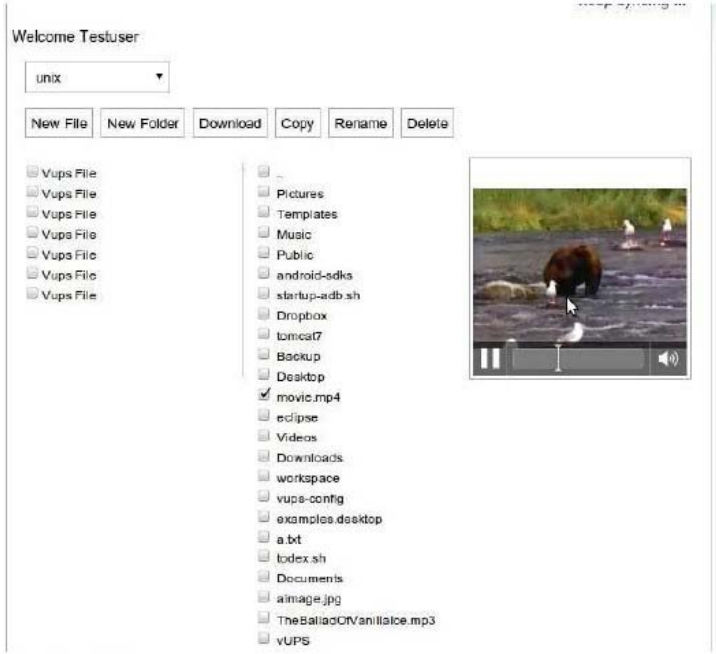
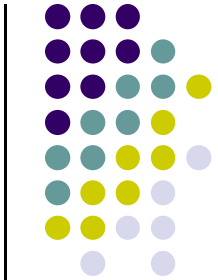


Fig. 3: vUPS User Interface

Performance Evaluation



- All the desktop machines have 8 GB memory and 2.7 GHz CPU. The laptop has 4 GB memory and 2.7 GHz CPU. The desktop machines run Windows 7, Ubuntu 11 and Mac operating systems, respectively. Using a Google Nexus One phone with 512 MB memory and 1 GHz CPU running Android 2.3 operating system. The DropBox account has 2 GB of storage.

File I/O Performance

- performance of file reading
  - a desktop is designated as the main terminal. Files of 1 KB, 10 KB, 50 KB, 100 KB, 200 KB, 500 KB, 1MB, and 3 MB are read by the smartphone via the main terminal

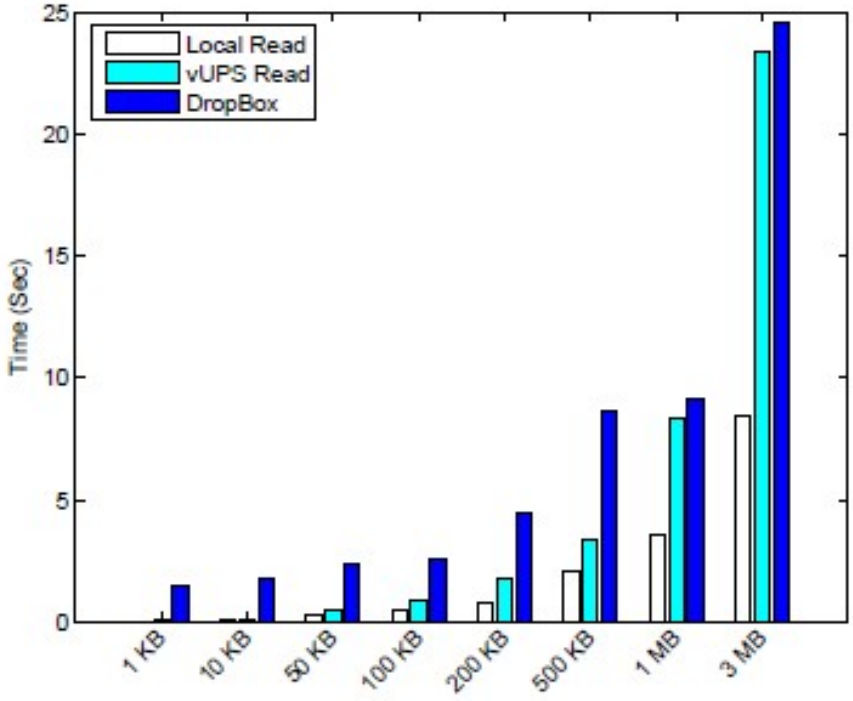


Fig. 4: File Size vs. Response Time

Performance of Metadata accesses

- For local file creation in the smartphone, creating 1000, 2000, 3000, and 4000 files using 1, 2, 3, and 4 threads

- Each thread first contacts the main terminal and requests to create

  1000 files in the device

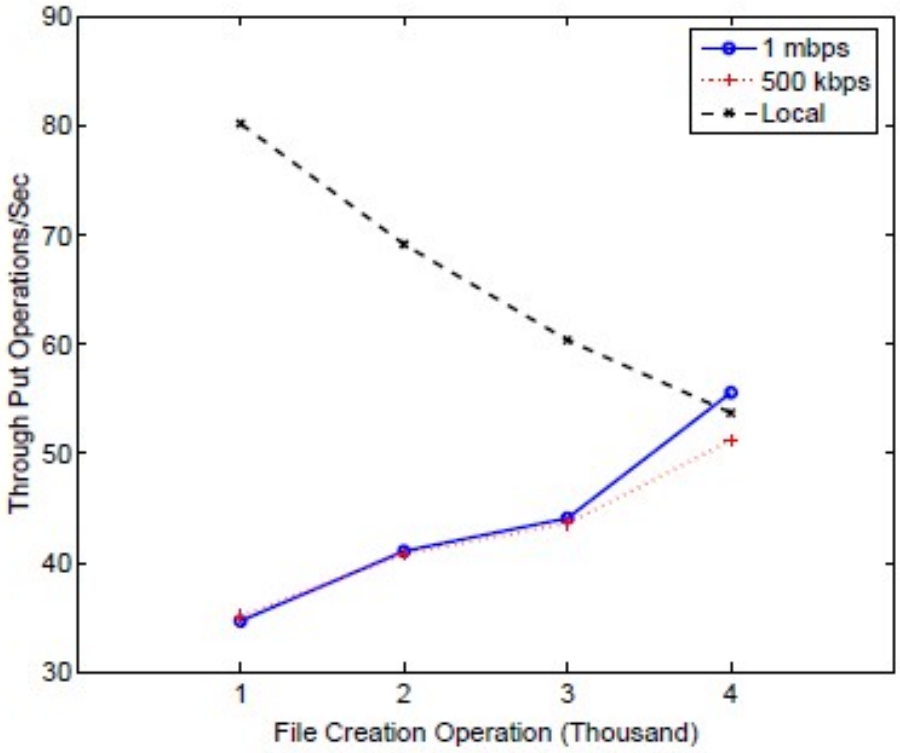


Fig. 7: Throughput of File Creation

Network Impact

- Run the mdtest file creation of 1000 files with varying network speed.
- The response time for 1000 file creation via the smartphone using one thread. The x-axis represents the network speed, the y-axis represents the system throughput.
- Conclusion: Improvement diminishes when the network speed is fast enough.

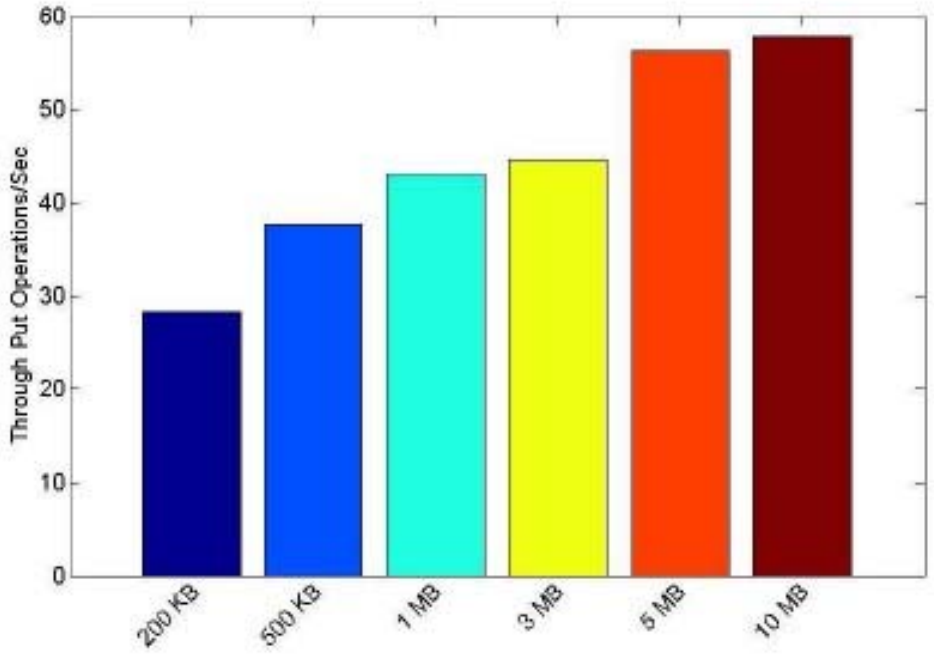


Fig. 8: Network Bandwidth vs File Creation

Network Impact

- Impact of network speed on file read operations
- The read throughput increases with the increase of bandwidth for vUPS while it remains stable for local read and DropBox.
- No constrain the bandwidth of the connection between the smartphone and the Dropbox

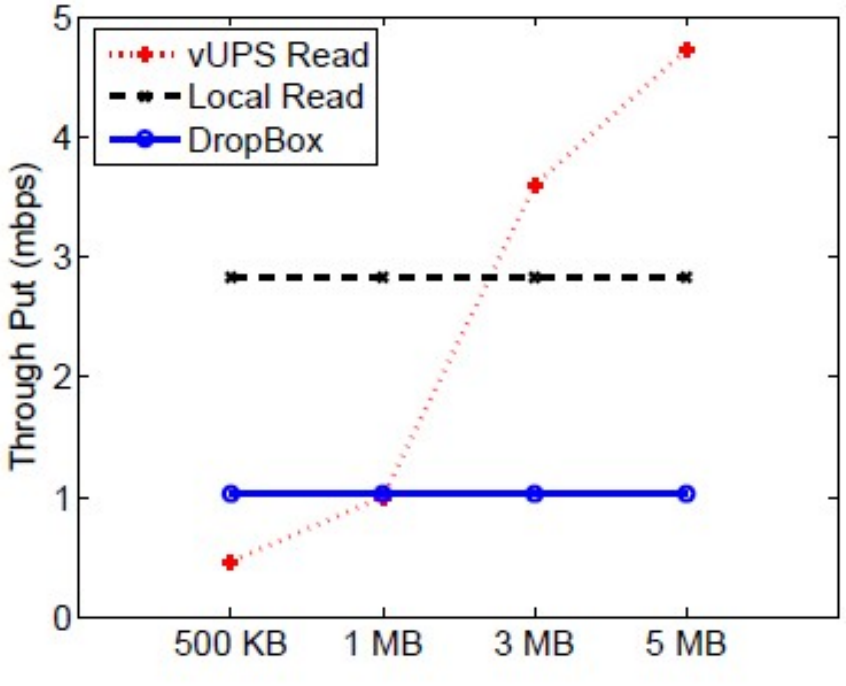


Fig. 9: Network Bandwidth vs File Read

Comparison to Dropbox

- The x-axis represents the number of files fetched randomly from these 50 files.
- The y-axis represents the response time
- DropBox first downloads all the files, and then accesses them randomly
- vUPS does not download all the files at first
- vUPS downloads the files when being accessed and then caches them for future references
- when the number of accessed files is larger, the performance of both vUPS and DropBox is similar to each other

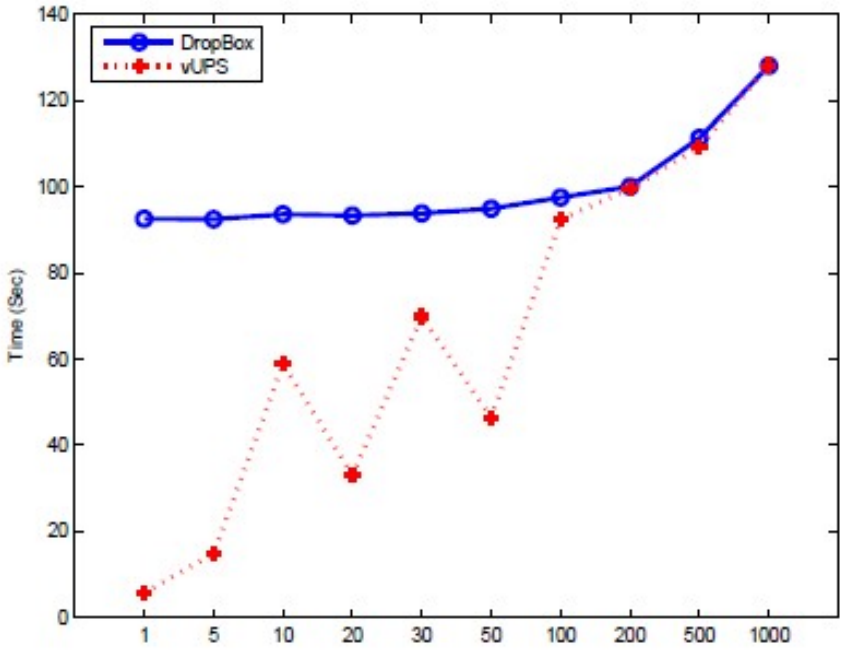
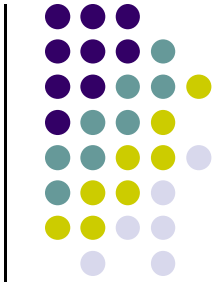
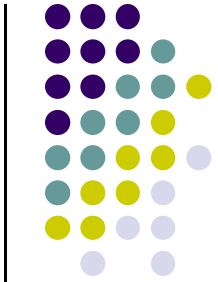


Fig. 10: DropBox vs vUPS

Conclusion:



- vUPS transparently and seamlessly integrate a user's personal storage space.
- A web interface is provided to the user with a global view of the files without involving any third party.
- Experiments based on the implemented prototype system show that vUPS can achieve similar user performance when compared to commodity commercial solutions such as DropBox.



- Thanks