Fast and Secure Laptop Backups

with Encrypted De-duplication

Le Zhang
<zhang.le@ed.ac.uk>
Paul Anderson
<dcspaul@ed.ac.uk>

LISA 2010

Laptop Backup Options

External Hard Drive

No offsite storage ?
What if I have a break-in? Or there is a fire?

I need a very large capacity to handle archival storage as well ...

Laptop Backup Options

DVDs are only small - I can only backups subsets of files ...

Recordable CD/DVD

I have to make multiple copies if I want offsite storage ...

Laptop Backup Options

Broadband upload speeds are slow - 30 DAYS to upload 300Gb to cloud storage is typical ...

Often, there is a transfer cost as well as a storage cost ...

Laptop Backup Options

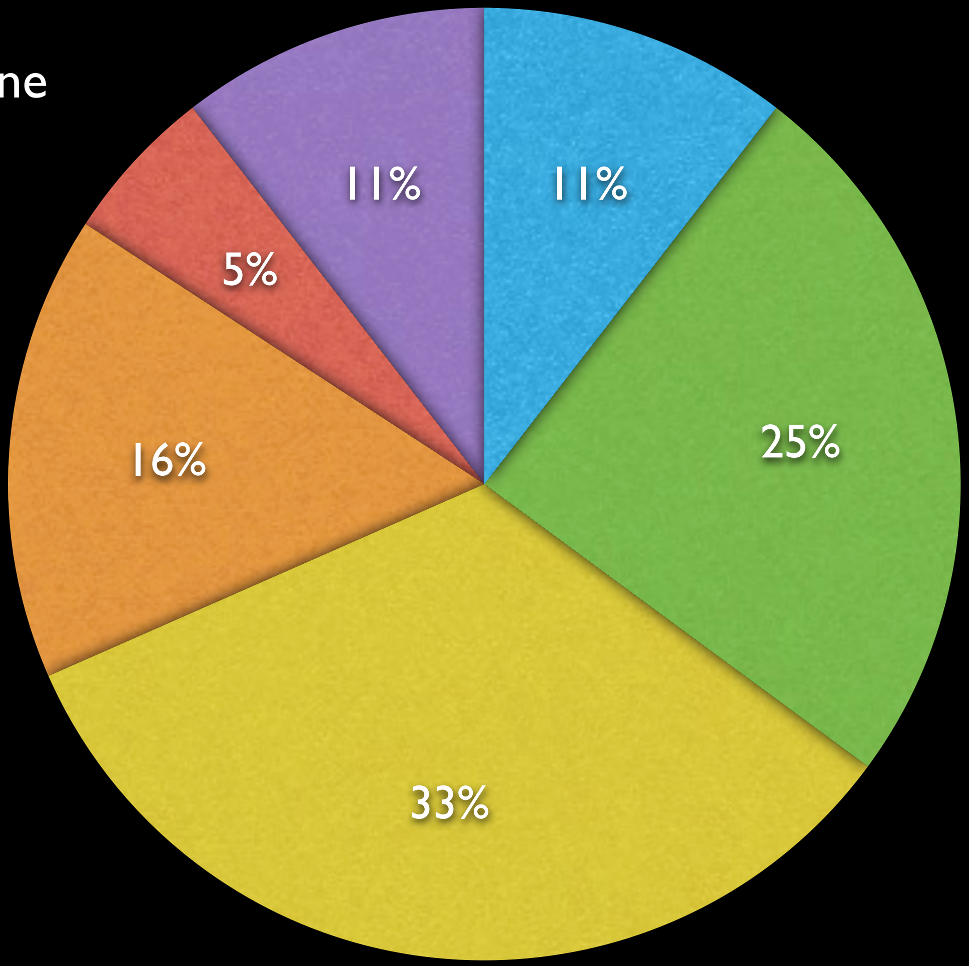
External Hard Drive

Recordable CD/DVD

Cloud Storage

What do people do?

- 🔵 Store no vital data
- 🟢 Regular full backups
- 🟡 Partial backups
- 🟠 Keep copy on University machine
- 🔴 Don’t do backups
- 🟣 Don’t use laptop

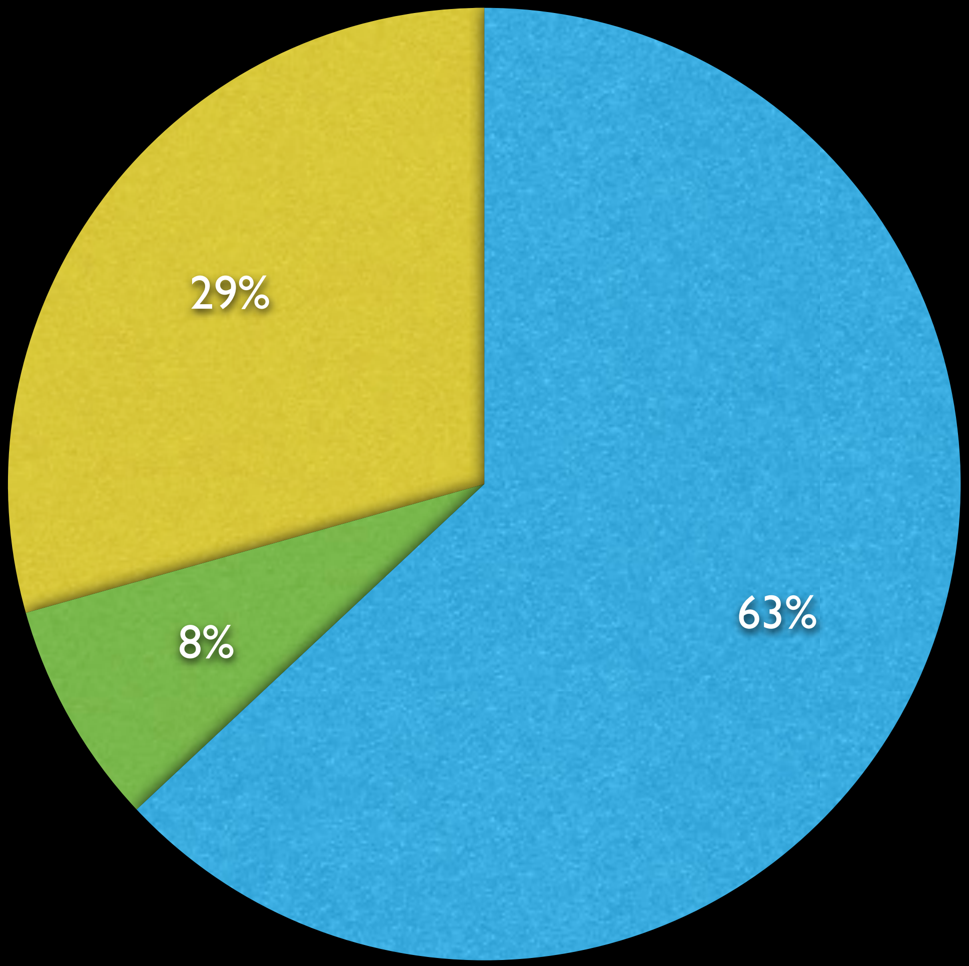


When people bother keeping backups, they are mostly ad-hoc - and usually only involve hand-selected subsets

What kind of data?

● User files
● Applications
● System files

Perhaps a lot of the system files and application files (at least) are common?



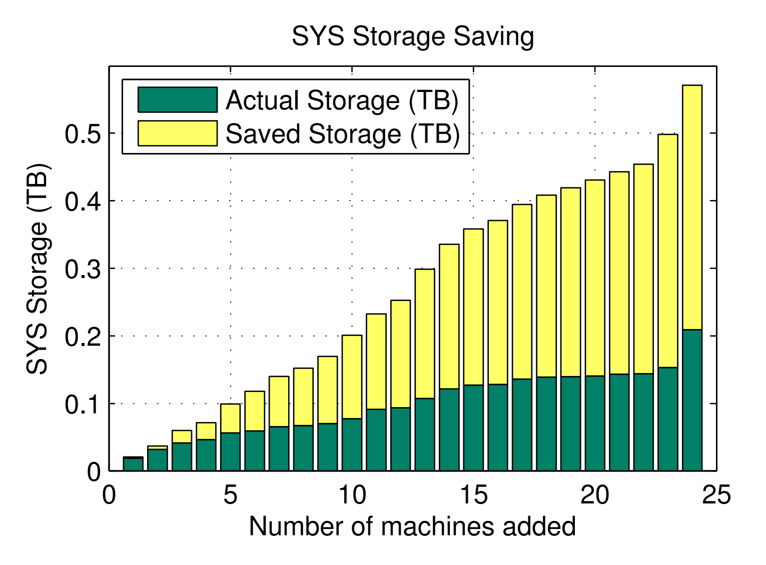
From our sample of academic Mac laptop users

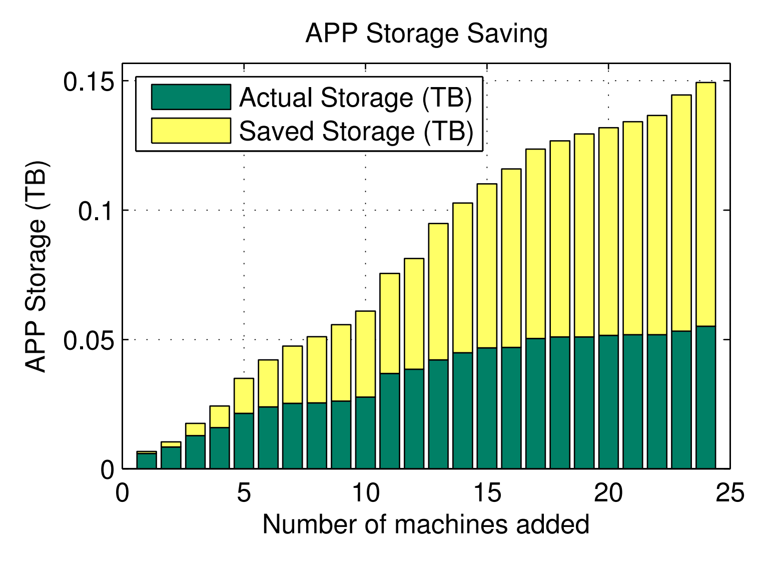
Shared Data

It seems like there is a good deal of duplication among the system and application files.

And this increases with the number of machines

But it is interesting that a good many files are not common! So is it a good idea not to back up these categories?

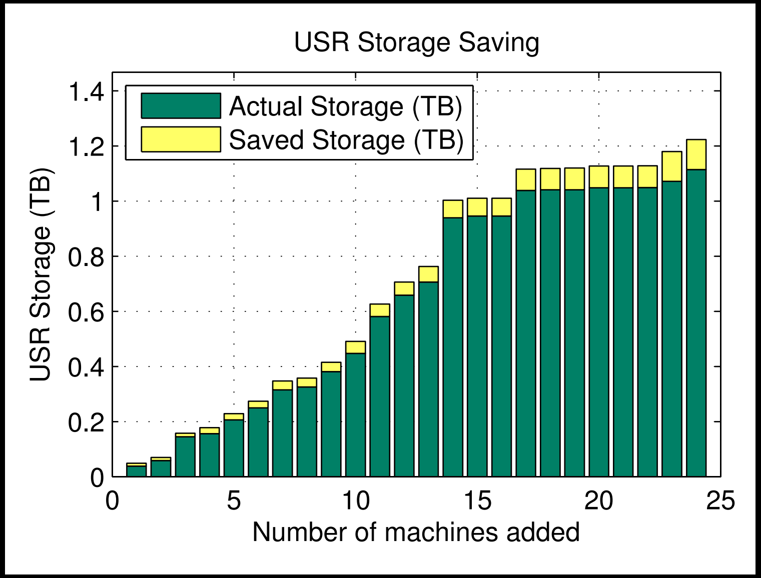


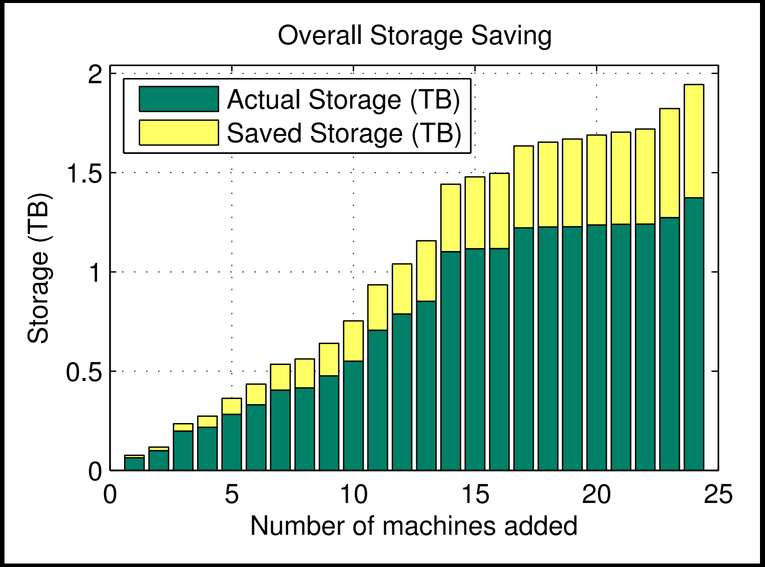


Shared Data

Obviously, there is less sharing among the user data - but the overall saving is still significant

And we might expect a higher degree of sharing among the user data for different communities - for example, common music files would make a big difference ...





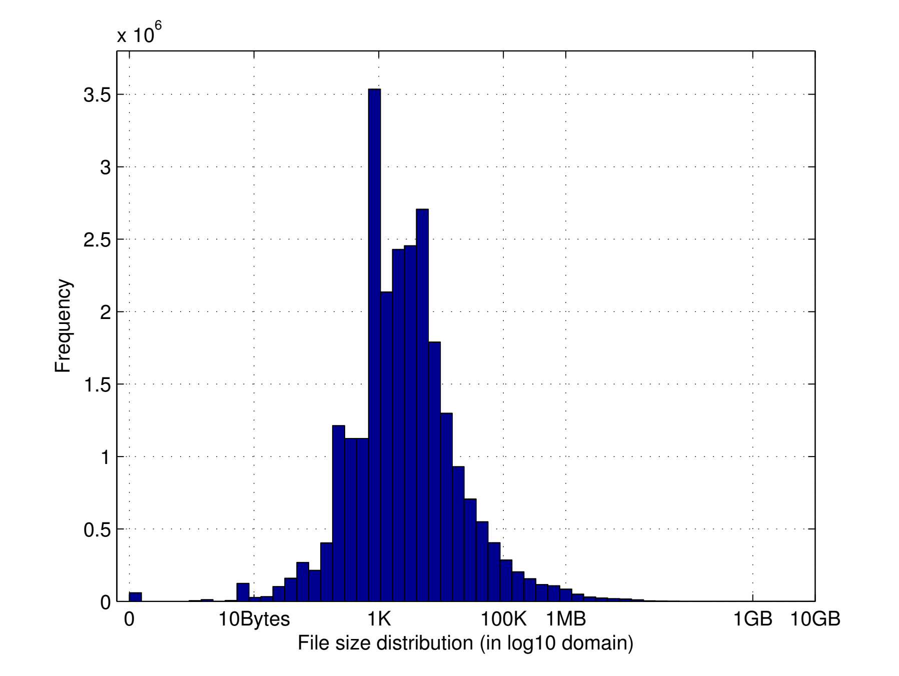
Deduplication

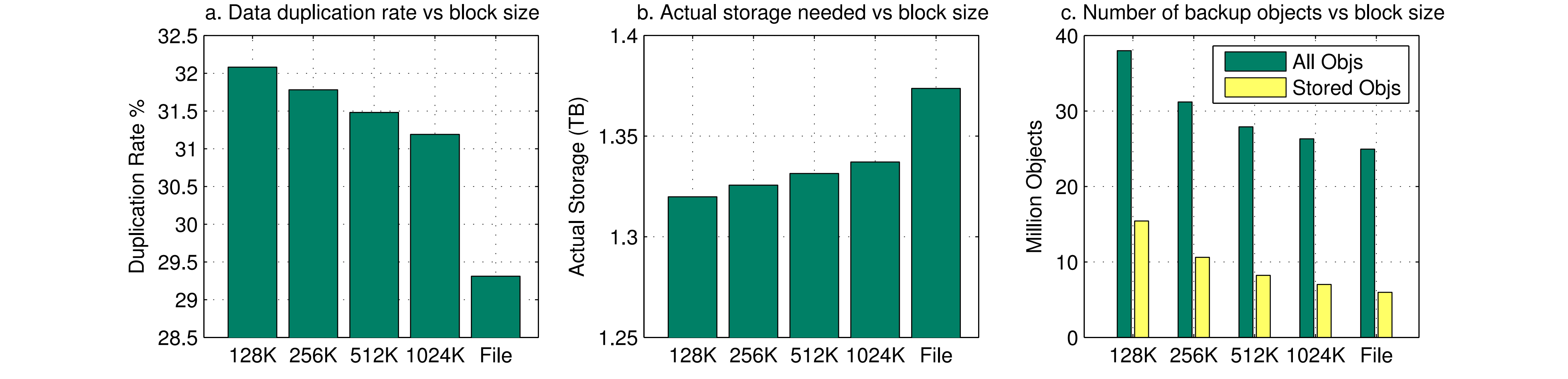
- “Deduplication” is becoming very popular for saving space when storing multiple copies of the same file
- A “hash” (digital signature) is generated from the contents of the file
- Two files with the same content will have the same hash
- Two files with different contents have a very high chance of having different hashes
- Use the hash as the name of the stored file

Block sizes

Deduplicating at the block level is more efficient than the file level.

What is an appropriate block size?




a. Data duplication rate vs block size

b. Actual storage needed vs block size

c. Number of backup objects vs block size

Deduplication problems?

- Most de-duplication systems work at the storage level
- This has two problems in our application ..
- If the data is encrypted “at source” (with different keys) then the deduplication is defeated (the cipher text will be different)
- The full data still has to be transmitted to the “server” - and this time is a more significant problem than the storage!

Convergent Encryption

- “Convergent Encryption” neatly solves the first problem ...
- Files are encrypted using the hash of the data as the key
- Files containing the same data will encrypt to the same cypher text and hence deduplication continues to work
- File owners will have the key (because they originally had the data) and will be able to decrypt the data - others won’t

Managing keys

- Each (unique) file now has a separate key which we need to manage
- Our solution creates a “data object” for each directory which contains the keys for the children, as well as their metadata
- The directory object is then encoded and stored in the same was as a normal file
- The user only has to record the key for the root object
- Entire duplicate subtrees can be detected

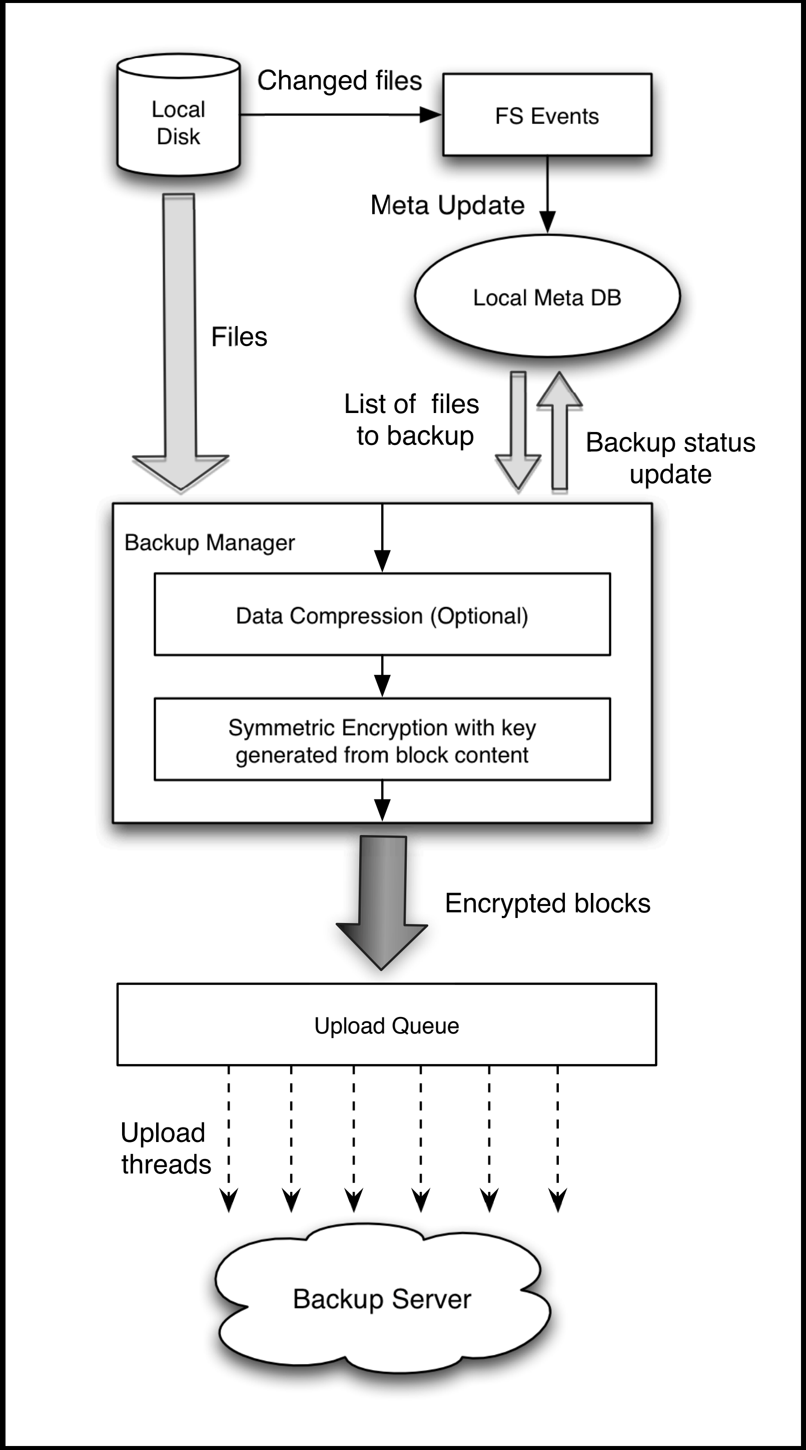
Avoiding Transmission

- To avoid transmitting data which already exists on the server, we need to do the deduplication on the source system
- Many services (eg. Amazon) don’t provide the necessary interfaces for the client to communicate directly
- There are several approaches to this, depending on specific application ...
  - A private server
  - A local “caching” server for a remote cloud service

A Protoype

A Mac OsX client

A local (departmental, home) server which performs hash checking, authentication and high-speed caching before forwarding unique blocks to the cloud



Where next?

- Performance depends heavily on the characteristics of the data itself, and the underlying network/storage (eg. latency)
    - We would like to study this more
- We would like to develop a production quality client, and investigate a possible service in a datacentre
    - we are looking for possible funding/partners

Fast and Secure Laptop Backups

with Encrypted De-duplication

Le Zhang
<zhang.le@ed.ac.uk>
Paul Anderson
<dcspaul@ed.ac.uk>

LISA 2010