Online Backup And Versioning In Log-Structured File Systems

Ravi Tandon[1]

[1]Computer Science And Engineering
Indian Institute of Technology Guwahati

HIPC Conference, 2012

Outline of Topics

Terms And Explanations

- Online Consistent Backup:
  The process of storing dependable versions of file system data is referred to as “backup“.
- Why Online Backup?
  System downtime may lead to enormous loses for enterprise applications.

Literature Review - Offline Backup

- Point In Time Copy[AFSM02]: Focus predominantly on the reduction of time required for the backup to complete. Consistency guaranteed in offline mode.
- Snapshot Mechanism: File systems snapshot file system state before copying the data. Eg. Andrew file system [HC88], Petal[LT96], Spiralog [GBD96] etc. Consistency of data is not guaranteed when the applications collide with the backup process.

Literature Review - Online Backup

- Online Backup Schemes: Employ *locking*, *detection of file movement* and *copy-on-write* [Shu91].
- Consistent Online Backup: Lipika Deka et.al. [DB10] propose *mutual serializability* for backup transactions. Aborting transactions that collide with backup process to ensure serialization is a major drawback.

System Model

The file system is assumed to have the following properties:

- Transactional system calls.
- Copy-on-write.

Protocol Definitions

- Post Backup Data.
- Conflict Dependency.
  - Conflict Depender Transaction
  - Conflict Dependee Transaction

Hypothesis

"The basic philosophy behind the design on OBVLFS is the identification of conflicting transactions. Backing up data written to file system by a conflicting transaction leads to inconsistency in file system state. The novelty in our design is the concurrent access of a file by a backup process and a conflicting transaction."

Protocol Overview

The backup process reads data written by non-conflicting transaction set. This requires the identification of conflicting transaction set. It is performed as follows:

- Identification of Conflicting Transactions.
  1. Incomplete Transactions.
  2. Post Backup Transactions.
  3. Transactions conflict dependent with a post backup transaction.
- Transaction Collision Prevention.

Conclusion

- An online backup scheme which circumvents transaction aborts and guarantees consistency.
- Transactions are classified into *conflict dependent* and *non-conflicting transactions*.
- Proposes a framework for constructing snapshots and storing them as multiple versions of file system's states.

Future work

- Formalization of *OBVLFS* to prove consistency.
- Implementation of backup scheme on a log-structured file system (preferably *YAFFS*).
- *Metadata compression* and *incremental versioning* to remove redundancy in data storage.

References


A. Azagury, MF Factor, J. Satran, and W. Micka, *Point-in-time copy: Yesterday, today and tomorrow*, NASA CONFERENCE PUBLICATION, NASA; 1998, 2002, pp. 259–270.


L. Deka and G. Barua, *On-line consistent backup in transactional file systems*, Proceedings of the first ACM asia-pacific workshop on Workshop on systems, ACM, 2010, pp. 37–42.


R.J. Green, A.C. Baird, and J.C. Davies, *Designing a fast, on-line backup system for a log-structured file system*, Digital Technical Journal **8** (1996), 32–45.


J.H. Howard and Carnegie-Mellon University. Information Technology Center, *An overview of the andrew file system*, Carnegie Mellon University, Information Technology Center, 1988.


E.K. Lee and C.A. Thekkath, *Petal: Distributed virtual disks*, ACM SIGOPS Operating Systems Review **30** (1996), no. 5, 84–92.


S. Shumway, *Issues in on-line backup*, Proceedings of the Fifth Large Installation Systems Administration Conference, 1991, pp. 81–87.