Venti: a new approach to archival storage[1]

Presented By: 陈子旸

Fudan University

13307130148@fudan.edu.cn

January 2, 2015

---

[1]powered by XƎLATEX

Outline

Venti

Presented By:
陈子旸

Outline

Abstract
Background

Venti

Data
Organiza-
tion[Organization]

Application
Example

Vac

1. **Outline**
   - **Abstract**
   - **Background**

2. Venti

3. Data Organization[Organization]

4. Application Example

Abstract

Venti

Presented By:
陈子旸

Outline
Abstract
Background
Venti
Data
Organiza-
tion[Organization]

Application
Example
Vac

- Venti: A network storage system intended for archival data
- A building block for a variety of storage applications
    1. logical backup
    2. physical backup
    3. snapshot file systems

- A block is identified by a unique hash of it's contents
- Enforce a write-once policy
- Duplicate copies of a block can be coalesced

Archival Storage

Venti

Presented By:
陈子旸

Outline
Abstract
Background
Venti
Data
Organiza-
tion[Organization]
Application
Example
Vac

- Purpose
  - Store data for long periods of time (forever)
  - Data may not be needed frequently, but when it is needed it is often crucial

Prevalent Form

Venti

Presented By:
陈子旸

Outline
Abstract
Background
Venti
Data
Organiza-
tion[Organization]
Application
Example
Vac

- Tape backup
    - Backup data to magnetic tape
    - (tar, ufsdump...)
    - Full backup vs Incremental backup
    - To provide backup as a central service for a number of client machines

Prevalent Form

- Snapshot
  - A snapshot is a consistent read-only view of the file system at some point in the past.
  - Each snapshot is a complete file system tree, much like a full backup.
  - A snapshot only requires additional storage for the blocks that have changed, like a incremental backup.
  - Always available and easy to access
  - Plan 9, WAFL, AFS…

Outline

1 Outline

2 Venti

3 Data Organization[Organization]

4 Application Example

Venti Archival Storage

- Goal: To provide a write-once archival reponsitory than can be shared by mutiple client machines and applications.
- Block level network storage system
  - Actually a backend storage for client apps
- Blocks addressed by hash of their contents
  - Use SHA-1 algorithm
  - Use hash value as its unique 'fingerprint'
- Write-Once policy
  - Block once written, never modified
  - Modified blocks will have new address

Why SHA-1?

- SHA-1 hash function is developed by NIST
- Output 160 bit hash values(20 bytes)
- Probability that there will be one or more collisions:

$$p \leq \frac{n(n-1)}{2} \times \frac{1}{2^b}$$

- Consider a large storage system contains 10^{18} byte of data stored as 8 Kbyte blocks($\sim 10^{14}$ blocks), the probability is less then 10^{-20}.
- Variants of SHA-1 can produce 256, 384 and 512 bit results for future use.

Venti Archival Storage

- Multiple clients can Share a Venti server
  - Hash function gives an unversal namespace
  - Duplication increases the utility rate of space
- Inherent integrity checking for data
- Caching is simplified
- Uses magnetic disk as storage technology
  - Access time comparable to non-archival data

Outline

Venti

Presented By:
陈子旸

Outline
Abstract
Background

Venti
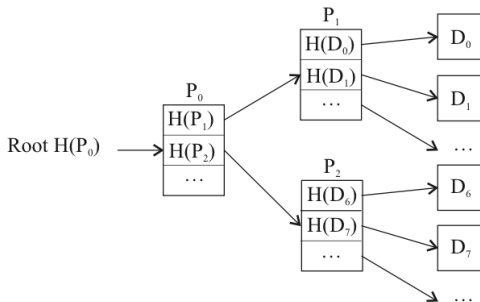
Data
Organiza-
tion[Organization]

Application
Example
Vac

Data Organization

- Data is divided into blocks and written to the server
- Pack the fingerprints into additional blocks, called pointer blocks, that are also written to the server
- Until a single fingerprint is obtained
- Applications can use such a structure to store a single file or to mimic the behavior of a physical device such as a tape or a disk drive

Figure 1. A tree structure for storing a linear sequence of blocks

- Venti does not allow such a tree to be modified
- But new versions of the tree can be generated efficiently by storing the new or modified data blocks and reusing the unchanged sections
- By mixing data and fingerprints in a block, more complex data structures can be constructed.
- For example, a structure for storing a file system may include three types of blocks:
  - Directory
  - Pointer
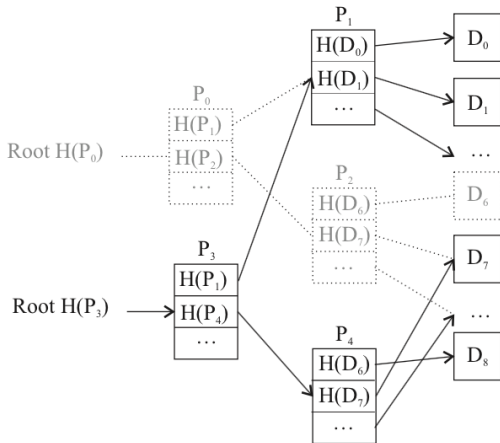 - Data.

DataOrganization

Figure 2. Build a new version of the tree.

Outline

Venti

Presented By:
陈子旸

Outline
Abstract
Background

Venti

Data
Organiza-
tion[Organization]

Application
Example
Vac

1. **Outline**

2. **Venti**

3. **Data Organization[Organization]**

4. **Application Example**
   - Vac

- Vac is an application similar to tar and zip
    - With vac, Selected files will be stored as a tree of blocks on Venti server.
    - The output is always 45 bytes long, included a 20 byte root fingerprint.
    - 'unvac' enables user to estore files from a vac archive.
- Vac writes each file as a seperated collection of Venti blocks, which can coalesce duplicate copies of a file
- Incremental backups options can improve performance