SNIA ■ SANTA CLARA, 2015

ZFS Async Replication Enhancements

Richard Morris
Principal Software Engineer, Oracle
Peter Cudhea
Principal Software Engineer, Oracle

Talk Outline – Learning Objectives

- High level understanding - how ZFS provides an efficient platform for async replication
  - incremental send contain only the data that's changed between snapshots
- Finding stability in the chaos – tension between what's stable in an archive and what isn't
  - decouple what's sent from what's changing
- Resolving significant constraints - why something simple turned out not to be not so simple

2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

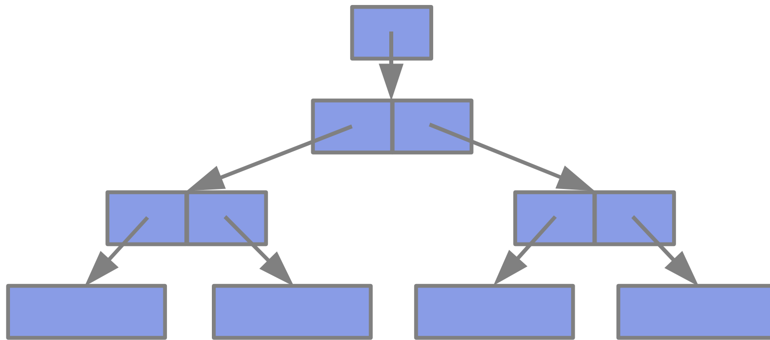
ZFS Overview

- Combined filesystem and volume manager
- Pooled storage
  - Does for storage what VM did for memory
- Scalable
- Block level compression and encryption
- Transactional object system
  - Atomic group commit
  - Always consistent on disk – no fsck
- Provable end-to-end data integrity

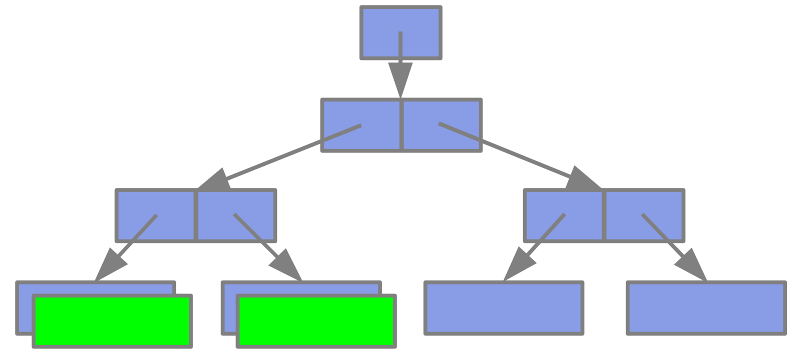
2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Copy-On-Write Transactions

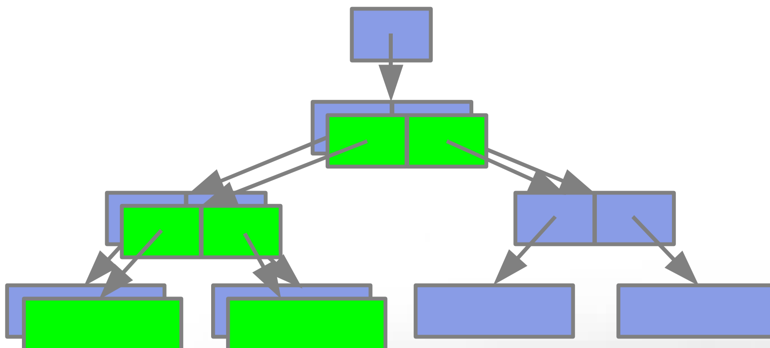
1. Initial block tree



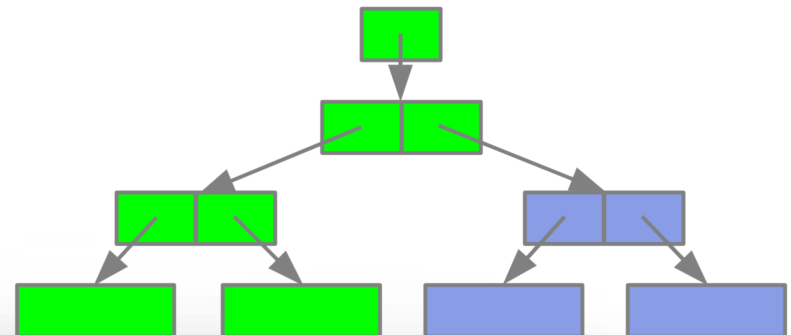
2. COW some blocks



3. COW indirect blocks



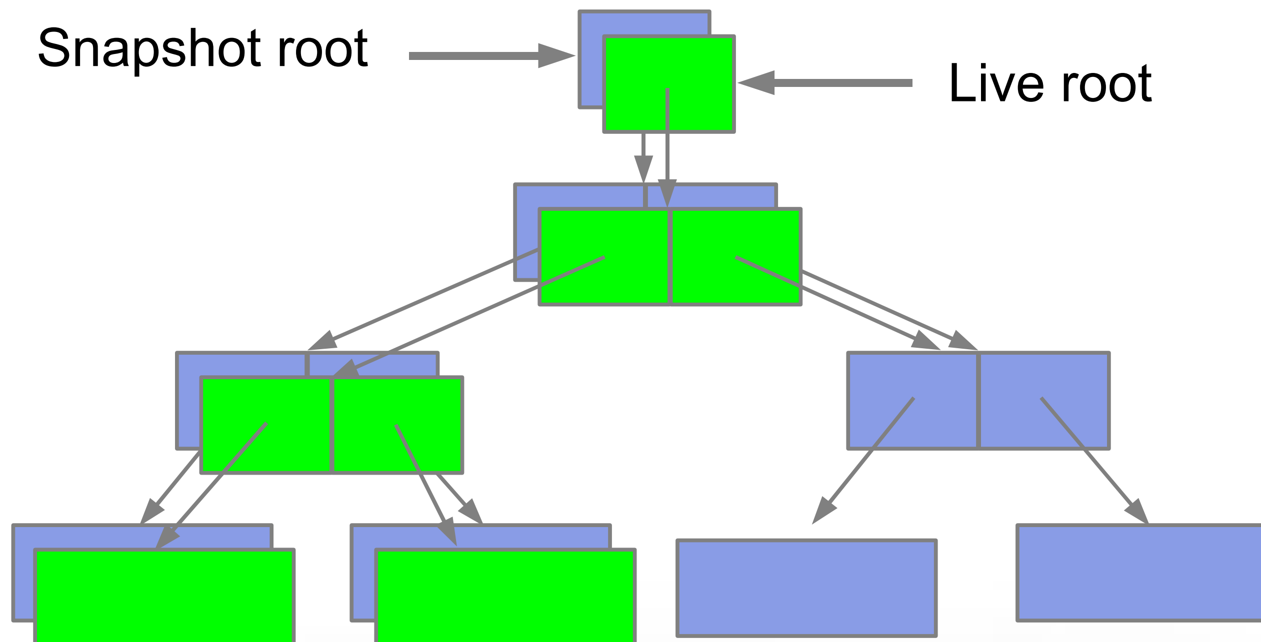
4. Rewrite uberblock (atomic)



SDC 2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Constant-Time Snapshots

- At end of TX group, don't free COWed blocks
  - Actually cheaper to take a snapshot than not!



2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

ZFS snapshot

- Read-only point-in-time copy of a file system
  - Instantaneous creation, unlimited number
  - No additional space used
  - Accessible through .zfs/snapshots in root of each file system
    - Allows users to recover files without sysadmin intervention
- Take a recursive snapshot of a home directory

```
# zfs snapshot -r home/richm@tues
```

- Rollback to a previous snapshot

```
# zfs rollback home/richm@mon
```

- Display the changes between two snapshots

```
# zfs diff home/richm@mon home/richm@tues
```


2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

ZFS send/receive

- unidirectional
  - only a limited backchannel available
- zfs send command (on source)
  - send the complete contents of a snapshot to stdout
  - send the incremental diffs from one snap to the next to stdout
- zfs recv command (on target)
  - create a snapshot from data read from stdin
- primary use case
  - to replicate data for backup or disaster recovery

2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Async Replication using ZFS send/recv (example)

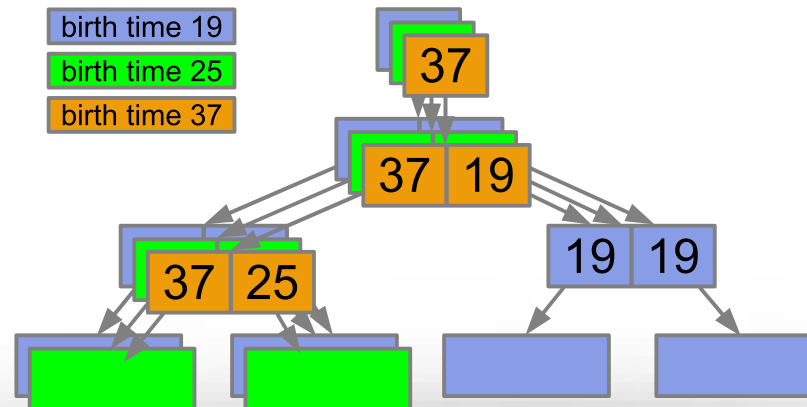
zfs send -r fs@monday | ssh target zfs recv backuploc

1) Serialize a dataset hierarchy at a point in time
2) Send the serialized stream to a backup location
3) Re-assemble the stream to duplicate the hierachy

2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Sending an incremental snapshot

- Goal: update the backup with changes from @monday to @tuesday
- sending side
  - traverse block tree of @tuesday
  - skip subtrees of blocks unchanged since @monday's birth time
- receiving side
  - apply the changes => so the backup is identical to @tuesday



2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Customer Issues with Replication

1) Transmission failures not detected right away
   - Stream contains limited checksum info
2) Compressed data is sent uncompressed
   - Uncompressing before sending wastes CPU, network bandwidth
3) Initial replication can take days to complete
   - Maybe be interrupted by a system failure, network outage, ...
   - Any failure requires replication be restarted from the beginning
   - Partially received data is thrown away

2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Issue 1: Failures may not be detected right away

Solution: additional checksums in the send stream

- New Stream Format
  - Each record in the stream has its own checksum
  - Guarantees only good data is saved to disk
  - Fail early if stream has degraded
  - Better protection from bitrot in archived streams
- Leverage existing per block checksums
- Add index to stream
  - Faster way to select only what's needed
  - Allows resuming from a fixed stream

2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Issue 2: Compressed data is sent uncompressed

Solution: send the compressed data

- Avoid dehydrating data for the wire
  - saves CPU and network bandwidth
  - if compression settings differ then target will make it right
- block checksum leveraged as the on-the-wire checksum
  - whenever on-disk checksum matches the on-the-wire data
- Encrypted data still decrypted and re-encrypted for now
  - key-present encrypted send+recv would be first step
  - keyless send and keyless receive possible but non-trivial!

2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Issue 3: A failure requires replication be restarted

Solution: Resumable Replication

- Detect failures as soon as possible then...
  - stop
  - resume from the point of interruption
    - skip over everything already sent
- Compatible with unidirectional model
- The content of each snapshot is stable
  - but the snapshot namespace is not

2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Resumable Replication – technical issues

1) Replicating changes to the snapshot namespace

- snapshots are being destroyed, renamed, cloned, ..
- prone to bugs and races
  - can be difficult to get a consistent view of namespace
  - changes to source namespace causes confusion in the stream
  - confusion in the stream causes confusion on the target

2) Verifying that all received data can be trusted

- All the saved data must be known to be good
- All the good data must be known to be saved

2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Issue 1: snapshot namespace keeps changing

Design options:

1) Full two-way communication (like rsync)
2) Stabilize entire source namespace prior to send
3) Stabilize only the snapshot namespace being sent

SDC 2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Issue 1: snapshot namespace keeps changing

Solution: Option 3 – stabilize and leverage the table of contents

- The TOC contains the list of snapshots in the stream
  - TOC is now authoritative
    - Creating the TOC is now close to atomic
  - the order in the TOC is the order on the wire
  - target side uses TOC to follow the namespace changes
  - other changes on source allowed but not propagated to target
  - therefore fewer inconsistencies need to be fixed on target

2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Issue 2: Verifying received data can be trusted

Solution: Resumable Chain of Custody

- Each record has a checksum => new stream format
- Records sent in monotonic order
- Partially received datasets saved persistently
- Receive Bookmark – how far previous receive got
- Resuming Send - restart from there
- Splice the resuming send into the resumable dataset

– Each of the above steps validate the received data and reject anything inconsistent

2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Issue 2: Verifying received data can be trusted

Resumable Chain of Custody

- Monotonic send order
  - The traverse order of the snapshot tree on source
    - Must match order of records on the wire
    - Must match birth time order of records on target
    - Required closing a few holes
- Receive Bookmark
  - Calculate the high water bookmark from target disk
    - Which object and offset changed most recently
    - Restart the send from that same bookmark
    - The resumed stream will fit right in with nothing missed

2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.

Conclusions

- ZFS provides an efficient platform for async replication
  - incremental snapshots contain only what's changed
- Stability in the chaos achieved by decoupling what's being sent from what's being changed
- Why something that seemed simple turned out not to be not so simple
  - needed to finish the job of making things stable

2015 Storage Developer Conference. © Insert Your Company Name. All Rights Reserved.