ZFS Backup Platform

ZFS Backup Platform

Robert Milkowski
Senior Systems Analyst
TalkTalk Group

http://milek.blogspot.com

The Problem

- Needed to add 100's new clients to backup
- But already run out of client licenses
- No spare capacity left (tapes, drives, …)
- Performance issues
- No money to spend

Traditional Backup Platforms

- EMC/Legato Networker
- Symantec/Veritas NetBackup
- Tivoli Storage Manager Server
- Open Source (Amanda, ...)
- tar, ufsdump, rsync, ...

Traditional Backup Platforms

- Usually licensed (expensive) per
  - client
  - backup/media server
  - tape library
- Skills (lack of)

Why Do We Need Them?

- Oracle/RMAN integration
- Integration with other 3rd party software
- Bare Metal Recovery
- Easy-of-use (???)
- Well known (skills)

Alternatives

- Open Source backup solutions
  - Cheap but too complicated
- In-house solution
  - Most flexible
  - Best use of latest technologies

General Idea

- Utilize commodity HW & open source
  - Each client assigned a filesystem
  - Data copied from a client into it
  - Snapshot created

Requirements

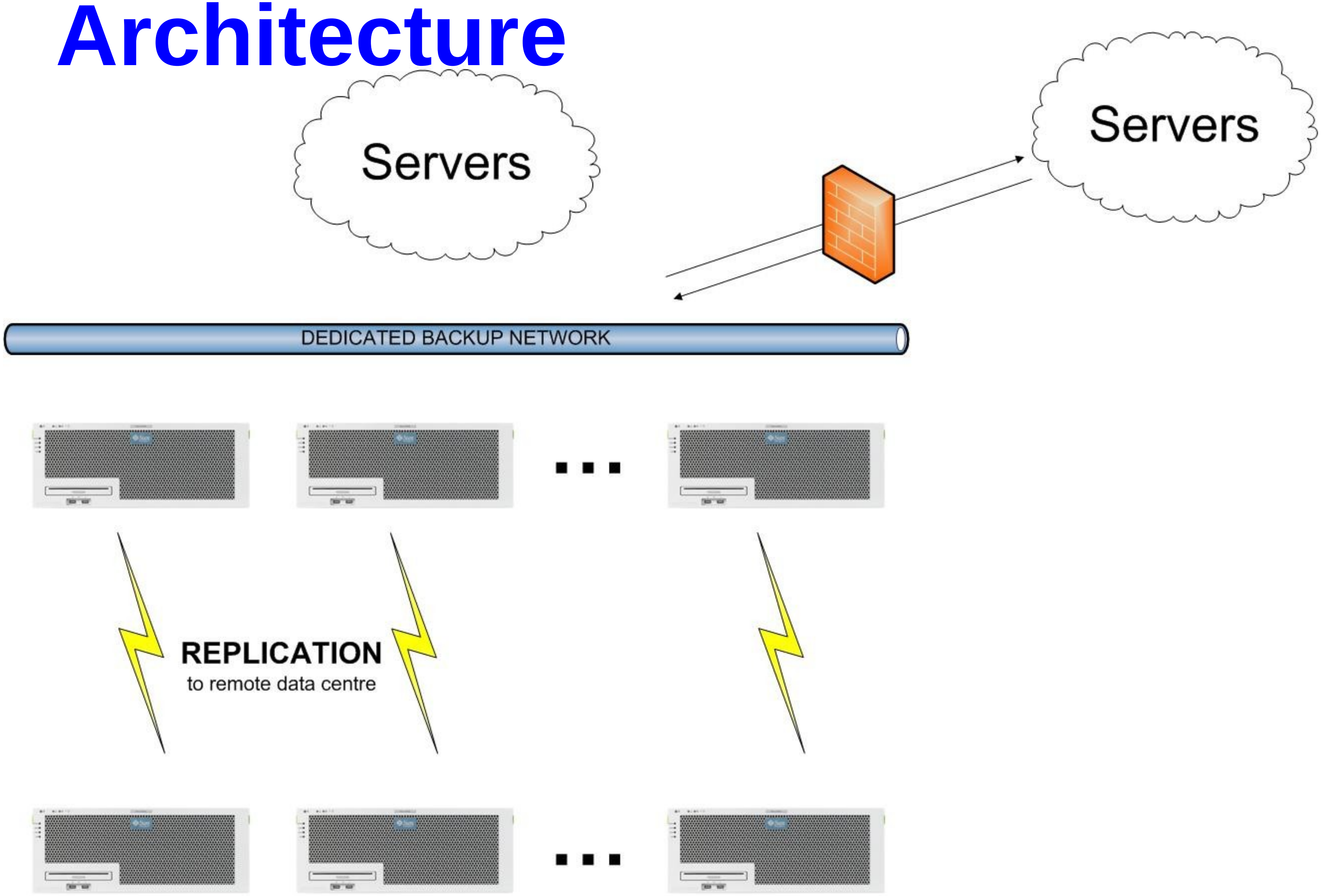
- Support different UNIX platforms
- **Significantly** cheaper
- Scalable to 1000s of clients
- Easy-to-use
- Remote Backup Copies

Requirements cont...

- Only well known and open source tools
- **Commodity hardware (x86, SATA)**
- Vendor neutral
- Horizontal scalability
- **A backup tool** to hide all the complexities

ZFS Backup Platform

Architecture



Storage Requirements

- Flexibility in disk space allocation
- Unlimited number of snapshots
- **Reliability**
- High sustain write throughput
- Easy storage management

Disk Based Backup Problem

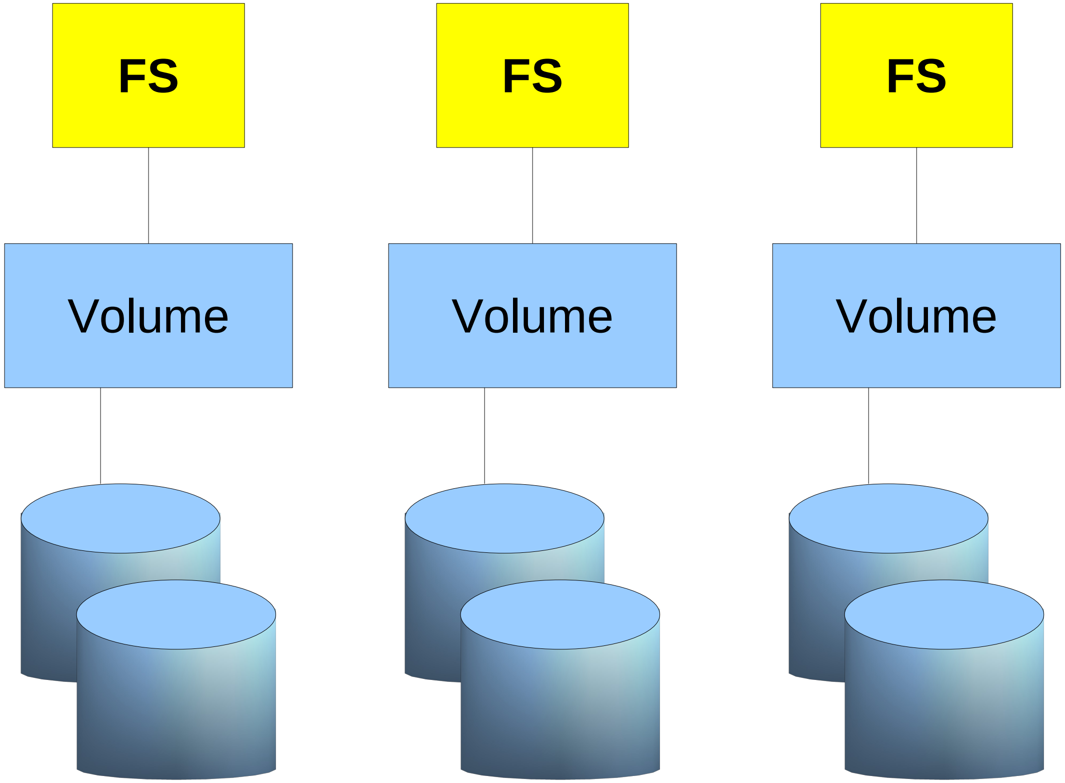
- **If a pool fails ALL backups are lost**
  - Dual Parity RAID
  - Hot Spares
  - Backups **replicated** to another node
  - Multiple backup nodes

Why ZFS?

- Dynamic filesystems and snapshots
- Incremental replication
- Built-in compression and dedup
- End-to-end data checksumming
- High write throughput
- Dual-parity RAID

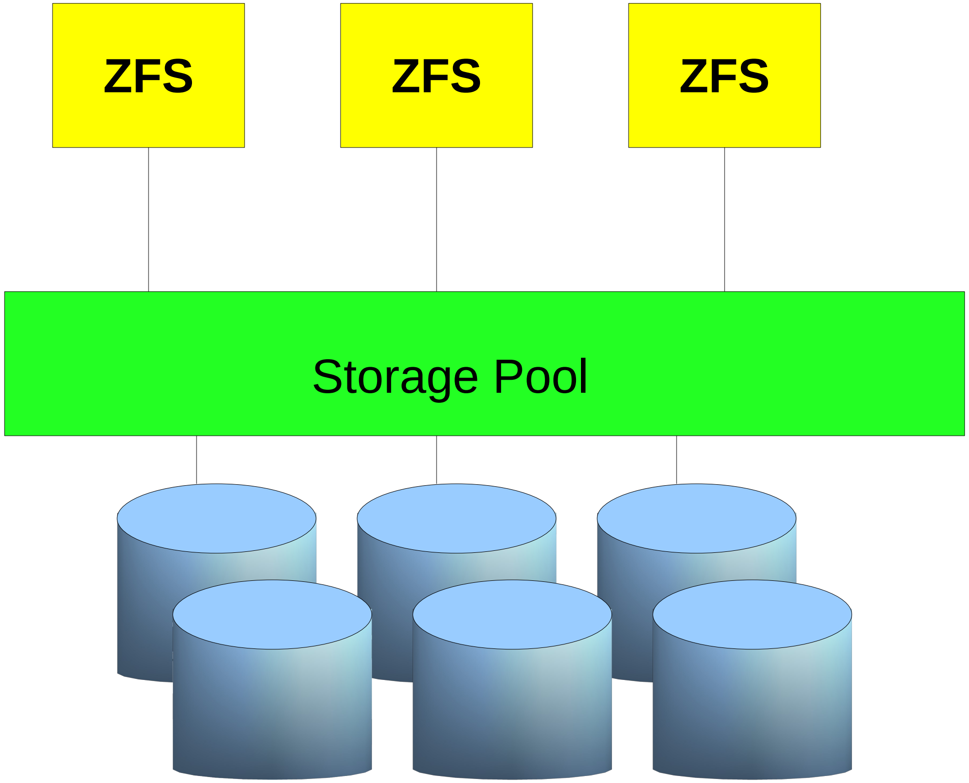
ZFS Backup Platform

Traditional FS+VM



ZFS Backup Platform

ZFS pooled storage



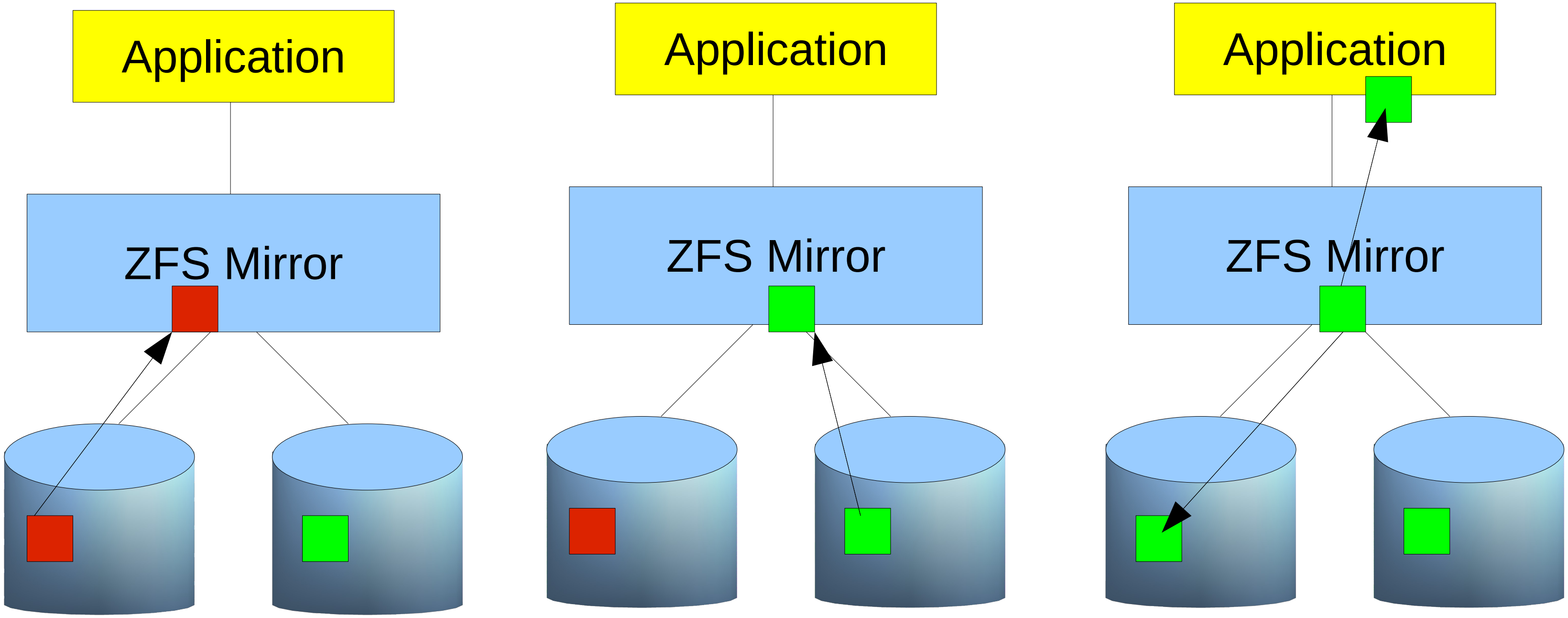
End-to-end data integrity

- Checksum checked after block is in memory
  - Whole IO path is checked
  - Corrects driver bugs, phantom writes, etc.
- Checksum and data block stored separately
  - Checksum is stored in parent block
  - Entire pool is self-validating
- Protects from accidental overwrites

Data integrity

- Both data and meta-data are checksumed
  - No silent data corruption
- Everything is Copy-On-Write
  - Never overwrite live data
  - Always consistent on disk
  - No need for fsck-like utility

ZFS Self Healing



Implementation Details

- Rsync daemon on each client
  - **The same** configuration on each
  - Only backup servers can connect
- Rsync initiated from a backup server
- **No extra** configuration on a client side

Implementation Details ...

1. ZFS filesystem created for each client
2. Data RSYNC'ed from the client
3. ZFS snapshot created

mk-archive-1.uk.intranet@backup-2009-10-30_16:25--2009-10-30_16:36

Implementation Details ...

- **Always incremental backups**
  - Yet all backups are full
  - Much smaller storage requirements
- LZJB compression enabled for all clients
- Deduplication enabled in the future
- Each backup accessible as RO filesystem

Implementation Details ...

```
pool-N/
  |-backup    all client backups are kept here
  |-logs      backup log files are there
  |-conf      configuration files
  |-scripts   tools
  |-archive   archives
  |-repl      replication area
```

Backup Tool

- Backups
- Archives
- Retention policies
- Reporting
- Replication
- ~~Restores~~

Backup Tool (cont.)

- Written in BASH
  - All sysadmins are familiar with it
- Easy to use
- All common operations implemented
  - backup, retention policies, archiving, ...

Backup Tool - Rsync

- RSYNC 3.x recommended
  - Partial filesystem listings
    - Much less memory consumed
- No ZFS ACL support

Inc/Excl Policies

- Global Incl/Excl policies
- Per-client Incl/Excl policies
- All configuration kept on a backup server

Retention Policies

- Global retention policy
- Per-client retention policy
- Deletes ZFS snapshots
- Does not apply to Archives

Client Replication

- Replicates all client backups
  - Based on zfs send|recv + mbuffer
- Global policy for archives and backups
- Per-client policy

```
$ backup -l -c mk-archive-1.uk.intranet
CLIENT NAME                               REFER   USED  RATIO  RETENTION    REPLICATE
mk-archive-1.uk.intranet                  65.5G  65.9G  2.15x  30  (global) no  (global)
```

Multiple Streams

- Helps to reduce a backup time
- Useful for clients with lots of small files
  - NFS appliances (latency)

```
$ backup -B -c mk-archive-1.uk.intranet -p 10
```

In-flight compression

- Helps to reduce a backup time
- Minimizes network usage
- Pushes more data than available bandwith
- Higher CPU impact on a client

```
$ backup -B -c mk-archive-1.uk.intranet -z
```

To backup a client

backup -B -c client [options]

backup -B -c client -r alternate_IP [opts]

```
$ backup -B -c mk-archive-1.uk.intranet
Using generic rules file: /archive-2/conf/standard-os.rsync.rules
Using client rules file: /archive-2/conf/mk-archive-1.uk.intranet.rsync.rules
Temporary log file: /archive-2/logs/mk-archive-1.uk.intranet/mk-archive-1.uk.intranet.rsync.2009-10-30_16:25
Starting backup
Creating snapshot archive-2/backup/mk-archive-1.uk.intranet@backup-2009-10-30_16:25--2009-10-30_16:26
Log file: /archive-2/logs/mk-archive-1.uk.intranet/mk-archive-1.uk.intranet.rsync.2009-10-30_16:25--2009-10-30_16:26
```

To list backups for a client

backup -lv -c client_name

```
$ backup -lv -c mk-archive-1.uk.intranet
CLIENT NAME                                                             REFER   USED  RATIO  RETENTION    REPLICATE
mk-archive-1.uk.intranet                                                2.58G  16.8G  2.48x  30  (global)    no  (global)
mk-archive-1.uk.intranet@backup-2009-09-30_07:00--2009-09-30_07:04      12.4G  7.30G  2.34x
mk-archive-1.uk.intranet@backup-2009-10-01_07:00--2009-10-01_07:01      5.30G   103M  2.97x
mk-archive-1.uk.intranet@backup-2009-10-02_07:00--2009-10-02_07:01      5.35G   139M  2.97x
[...]
mk-archive-1.uk.intranet@backup-2009-10-27_07:00--2009-10-27_07:01      2.60G   110M  1.74x
mk-archive-1.uk.intranet@backup-2009-10-28_07:00--2009-10-28_07:01      2.61G   124M  1.75x
mk-archive-1.uk.intranet@backup-2009-10-29_07:00--2009-10-29_07:01      2.61G   124M  1.75x
mk-archive-1.uk.intranet@backup-2009-10-30_07:00--2009-10-30_07:01      2.62G   132M  1.75x
mk-archive-1.uk.intranet@backup-2009-10-30_16:25--2009-10-30_16:26      2.58G      0  1.74x
```

Restore a file

```
# cd /archive-2/backup/mk-archive-1.uk.intranet/.zfs/snapshot
# ls | head -5
backup-2009-11-16_07:00--2009-11-16_07:01
backup-2009-11-17_07:00--2009-11-17_07:01
backup-2009-11-18_07:00--2009-11-18_07:01
backup-2009-11-19_07:00--2009-11-19_07:01
backup-2009-11-20_07:00--2009-11-20_07:01
#
# cat backup-2009-11-16_07:00--2009-11-16_07:01/etc/release
                       OpenSolaris Development snv_123 X86
           Copyright 2009 Sun Microsystems, Inc.  All Rights Reserved.
                        Use is subject to license terms.
                           Assembled 11 September 2009
#
```

Now use scp, cp, tar, rsync, ...

Example Report

```
$ backup -R yesterday

     Summary Report of Backups for 2009-12-14

Total number of clients in backup       :   221
Number of backups                       :   227
Number of failed backups                :     6
Number of successful backups            :   221
Number of clients with no backup       :     0
```

ZFS Backup Platform

backup -h

```
$ backup -h

usage: backup -B -c client_name [-r rsync_destination] [-z] [-hvq] [-a] [-p N]
       backup -l [-c client_name] [-v[F]] [-a]
       backup -L [-v[v]] [-c client_name]
       backup -R date [-v]
       backup -E [-v] [-n] [-c client_name]
       backup -e days -c client_name
       backup -m policy -c client_name [-a]
       backup -D backup_name [-f] [-a]
       backup -A -c client_name [-n] [-f] [-ff]
       backup -W -c client_name [-n] [-f] [-ff] [-a]
[...]
```

Nice to Have – TODO

- Centralized management / GUI
- Host groups and group schedules
- More sophisticated job scheduling
- Better reporting
- Restores

ZFS Backup Platform

Deployed Backup Nodes

- Sun x4500 servers, Open Solaris (snv_123)
  - 48x 750GB SATA disk drives
    - 4x 11 RAID-6 groups in one ZFS pool
    - 2x Hot Spare, 2x OS disks (mirrored)
    - 4x on-board GbE (802.3ad link aggr)
  - ~600MB/s sustain write throughput (pool)

Current Status

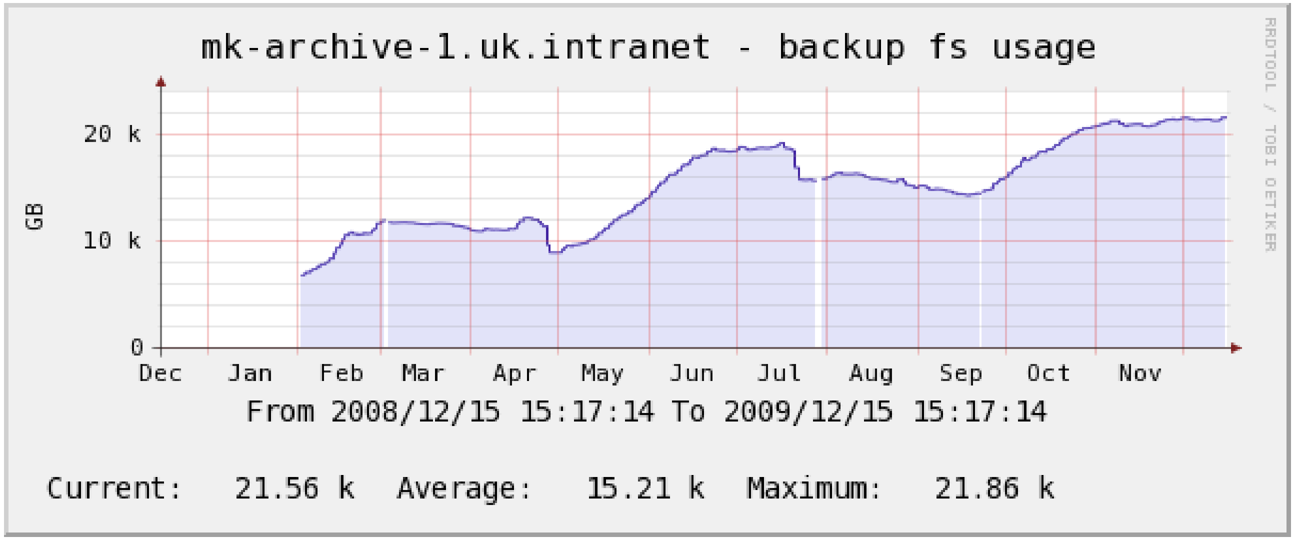
- Some clients replicated between nodes
- All archives replicated between nodes
  - ~400 clients
  - ~13,000 backups
  - <2% failed backups
  - ~40TB in-use
  - ~60TB un-compressed
  - ~6TB free

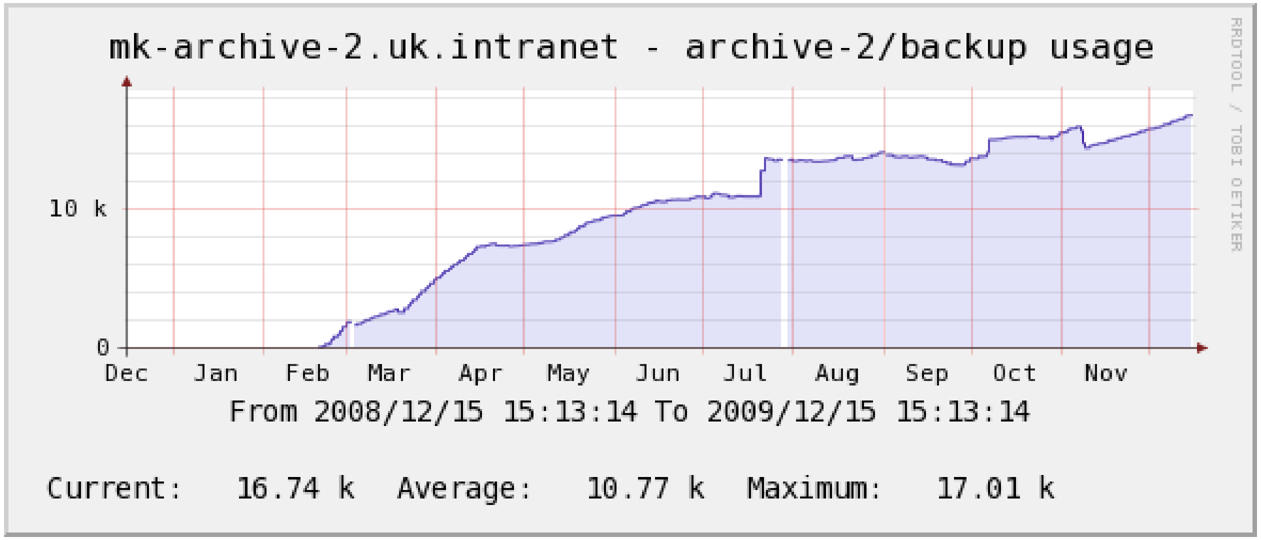
Current Status

- Archive-1 95% utilized
  - Very close to saturate available IOPS
  - Dedup might help or make it worse
- Archive-2 85% utilized
  - Much more head-room available
    - Less clients and data

ZFS Backup Platform

Historical Disk Space Usage





Current Status - Clients

Open Solaris

FreeBSD

Solaris

Linux

AIX

Summary

- Less storage required
- Less rack space required
- Less network bandwith utilization
- Quickier client backups
- Easy (and free) to use latest features

Summary ...

- Much more **cost effective**
- Proved to **scale** very well
- Easier to manage (less issues)
- Easier to use it
- **No hidden costs**

Useful links

http://milek.blogspot.com/2009/02/disruptive-backup-platform.html
http://milek.blogspot.com/2009/02/backup-tool.html
http://www.opensolaris.org/os/project/losug/files/June2009/Open_Backup_with_Notes.pdf
http://wikis.sun.com/display/BigAdmin/How+to+use+ZFS+and+rsync+to+create+a+backup+solution+with+versioning

http://opensolaris.org/os/community/zfs/

ZFS Backup Platform

Q&A

FSCK YOU!

if you use other filesystem than ZFS

:)))))))))

ZFS Backup Platform

Robert Milkowski
Senior Systems Analyst
TalkTalk Group

http://milek.blogspot.com

Auditing

- Easy way of comparing files
  - between backups or
  - between different clients
- BART, TRIPWIRE, diff, ...