Recent Filesystem Optimisations in FreeBSD

Ian Dowse and David Malone

15 June 2002

The Plan

- Review of Optimisations
- Benchmarking
- Analysis
- Results
- Conclusions/Future Work

Softupdates

Problem: Achieving async speed while keeping on-disk filesystem metadata recoverably consistent.

Solution: Reorder and sequence writes to allow async but maintain consistency.

Pros & Cons: Create/remove/extend $\Rightarrow$ win.

`fsync` semantics maintained.

Some implementation issues remain.

Dirpref

Problem: Long seeks between parent and child directories.

Solution: Bias allocation to place related directories close together.

Pros & Cons: Win for lots of directory traversal.

Possible issue with full disks?

Vmiodir

Problem: Directories cached in limited malloced memory.

Solution: Use the VM system instead.

Pros & Cons: Large directory working set OK.

Directories and files on equal footing.

Wasteful for small directories.

Dirhash

Problem: Directory lookups use a linear search.

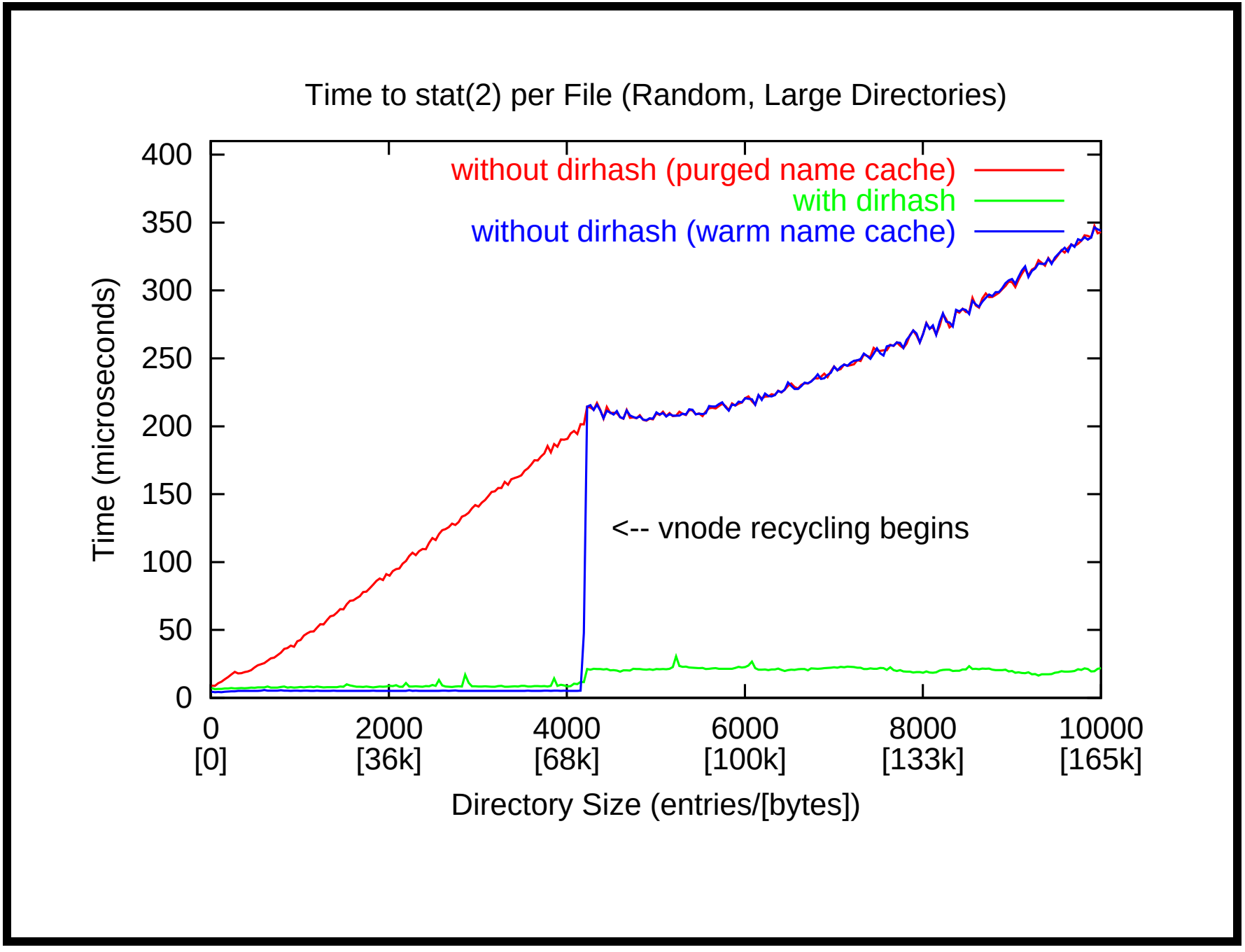
Solution: Build in-core hash table for directories when first accessed.

Pros & Cons: Win when you repeatedly access directories with lots of entries.

Pessimisation if directory is not accessed again.

Dirhash details

- Augments existing namecache.
- Hash built on first access.
- Also free space stats.
- m Random lookups from $O(nm)$ to $O(m+n)$.
- Should be easy to port.


Time to stat(2) per File (Random, Large Directories)
without dirhash (purged name cache)
with dirhash
without dirhash (warm name cache)
<-- vnode recycling begins
Time (microseconds)
400
350
300
250
200
150
100
50
0
0
[0]
2000
[36k]
4000
[68k]
6000
[100k]
8000
[133k]
10000
[165k]
Directory Size (entries/[bytes])

Testimonial

X11 Tar File:

Unpack: 300s $\xrightarrow{su}$ 90s $\xrightarrow{dp}$ 40s.

Find: 17s $\xrightarrow{dp}$ 3s $\xrightarrow{su}$ 4s.

Rm: 230s $\xrightarrow{su}$ 15s $\xrightarrow{dp}$ 4s.

33164 MH Mailbox:

Create: 815s $\xrightarrow{su}$ 30s $\xrightarrow{dh}$ 2.4s.

Pack: 1200s $\xrightarrow{su}$ 95s $\xrightarrow{dh}$ 2.4s.

Remove: 370s $\xrightarrow{su}$ 5s $\xrightarrow{dh}$ 1.4s.

Benchmarks

- Bonnie++
- Andrew ($\times 100$)
- Postmark
- Netnews
- Buildworld

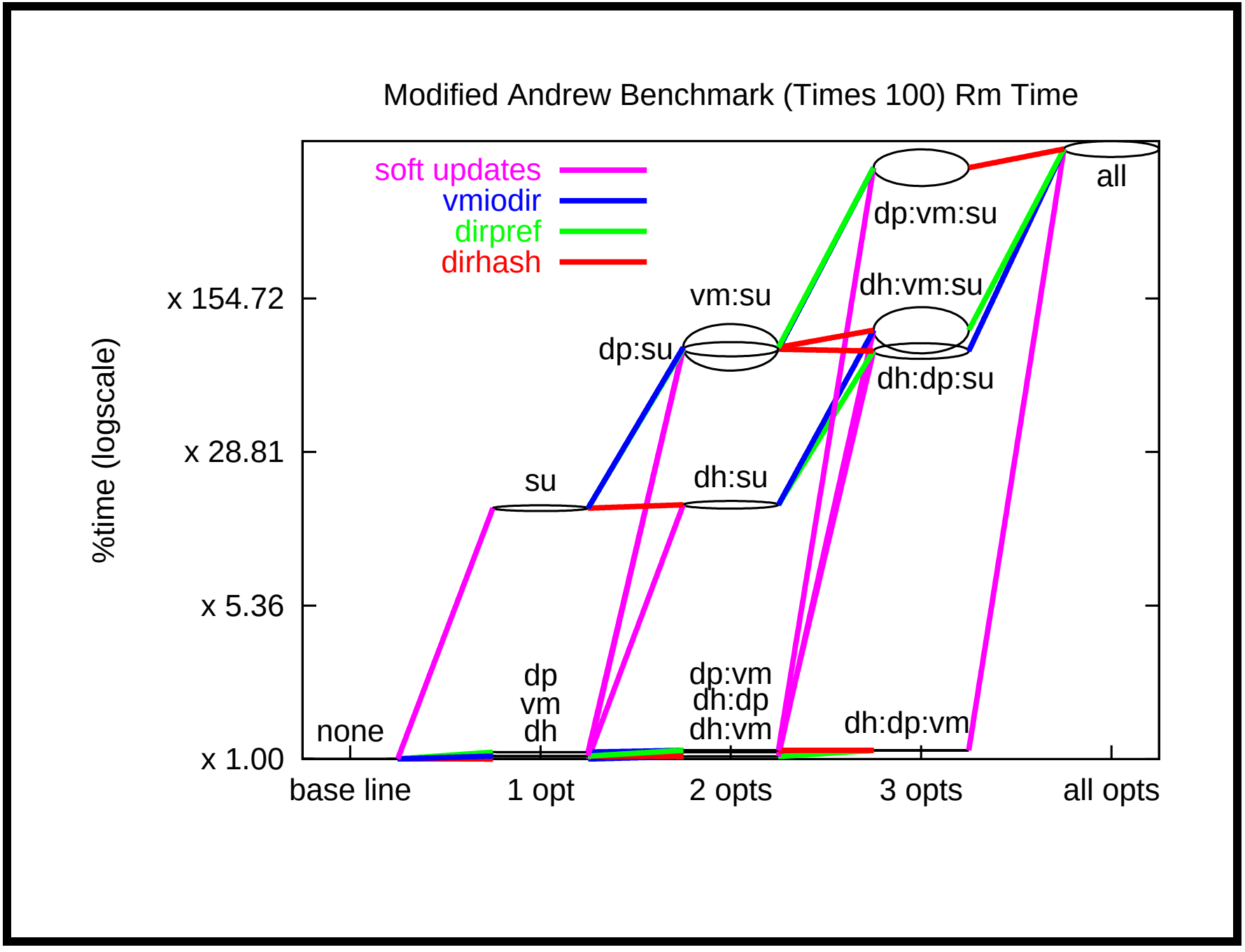
Method

- several runs of 16 combinations,
- `sync` and `rm` between runs,
- on slightly used `/usr` (aging?),
- 1.6GHz P4, 256MB ram, 20GB IDE disk, FreeBSD-4.5.

Now what?

Analysis

- 5 dimensional data,
- interactions of interest,
- normalise on all off,
- tables, linear models and plots.


Modified Andrew Benchmark (Times 100) Rm Time
soft updates
vmiodir
dirpref
dirhash
%time (logscale)
x 154.72
x 28.81
x 5.36
x 1.00
none
su
dp:su
vm:su
dp:vm:su
dh:vm:su
all
dh:su
dh:dp:su
dp
vm
dh
dp:vm
dh:dp
dh:vm
dh:dp:vm
base line
1 opt
2 opts
3 opts
all opts

Results

- Most improvements $\times 2 - \times 10$,
- Some around $\times 500$!
- Softupdates most significant,
- Dirpref and vmiodir overlap,
- Dirhash good for large dir churn.

Conclusions

- Dramatic improvements for testimonials and benchmarks.
- Very few pessimisations.
- Optimisations usually interact positively.
- Need to watch impact of layout policy changes.