HEAnet
IRELAND'S NATIONAL EDUCATION & RESEARCH NETWORK

Scaling Apache 2.x to > 50,000 users

colm.maccarthaigh@heanet.ie
colm@apache.org


CORDEBUS.HEANET.IE
FTP.HEANET.IE
PowerEdge
7250
DELL
PowerVault
2205

Material

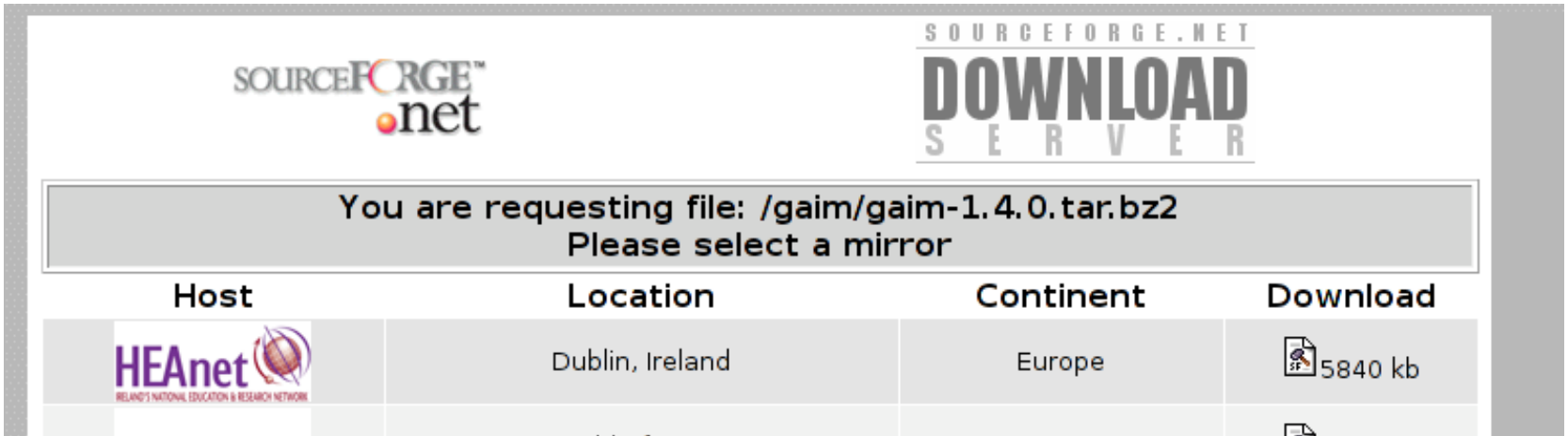
- Introduction and Overview
- Benchmarking
- Tuning Apache
- Tuning an Operating System
- Architectures

ftp.heanet.ie

- National Mirror Server for Ireland
  - http://ftp.heanet.ie/about/
  - http://ftp.heanet.ie/status/
- Used for Network/Systems Development
  - IPv6, Jumboframes, Multicast, etc
  - Apache 2.0/2.1/2.2

Mirror for

- Apache, Sourceforge, Debian, FreeBSD, RedHat, Fedora, Slackware, Ubuntu, NASA Worldwinds, Mandrake, SuSe, Gentoo, Linux, OpenBSD, NetBSD ... and much much more.



SOURCEFORGE.net

SOURCEFORGE.NET DOWNLOAD SERVER

You are requesting file: /gaim/gaim-1.4.0.tar.bz2
Please select a mirror

Host	Location	Continent	Download
HEAnet	Dublin, Ireland	Europe	5840 kb

The Numbers

- Roughly 27,000 concurrent downloads.
- 1200 Mbit/sec in production.
- 4Gbit/sec in testing.
- Roughly 80% of all Sourceforge downloads from April 2003 to April 2004.
- Usually 4 times busier than ftp.kernel.org

The numbers: a day

- 12,287,283 files stored
- 6.38 Terabytes of content available
- 5,498,725 downloads
- 5.93 Terabytes of content shipped

Resources

http://www.kegel.com/c10k.html

http://httpd.apache.org/

http://www.csn.ul.ie/~mel/projects/vm/

http://www.stdlib.net/~colmmacc/

Kernel sources

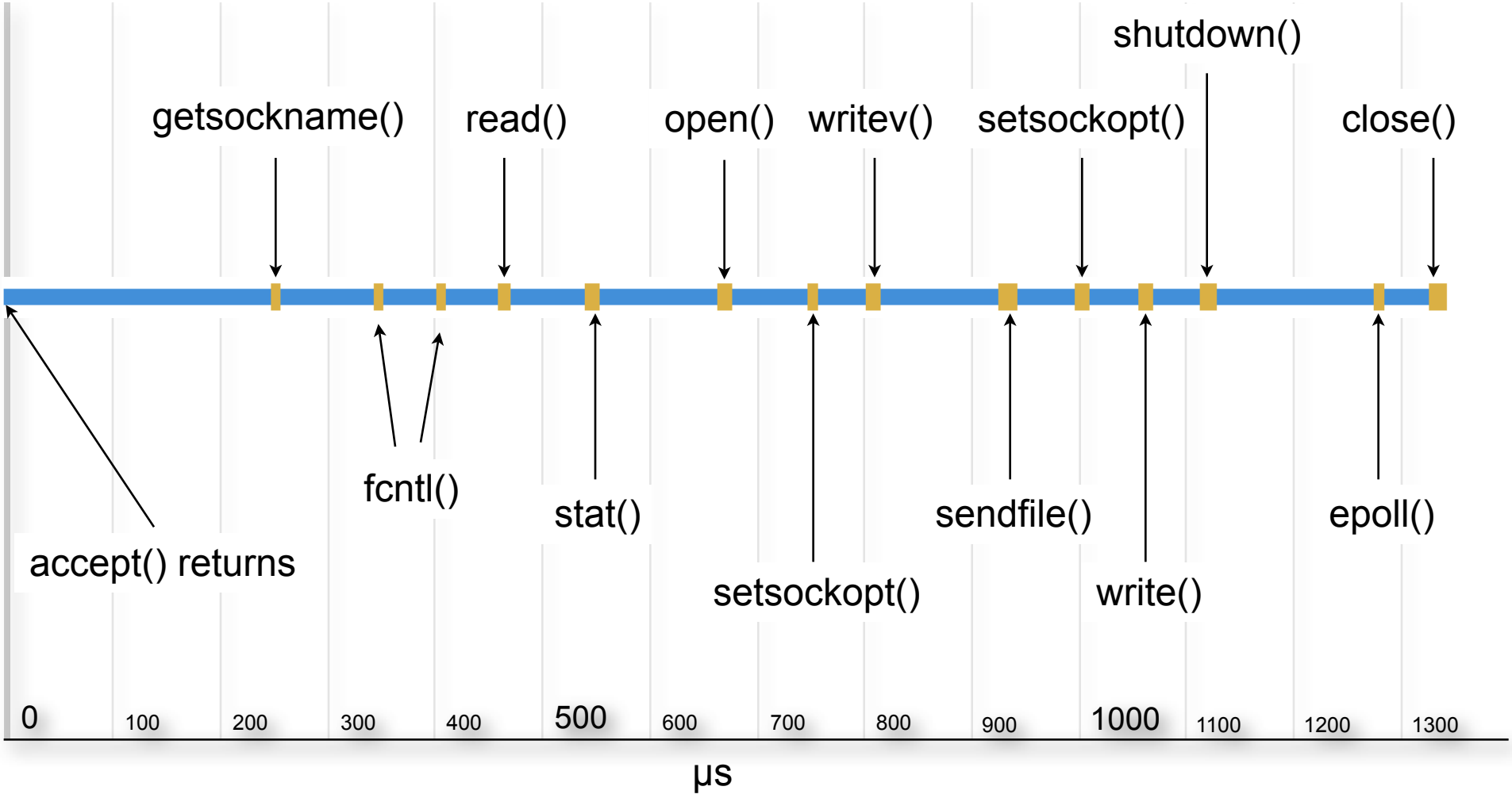
Tuning/NFS/high-availability HOWTO's

"Performance Tuning for Linux Server"

Web-server from a mile high

1. listen() + fork() / thread_create()
2. poll()/select()/epoll()/kqueue()/port_get()
3. accept()
4. stat(), readdir(), stat()
5. read() + write() / sendfile()
6. close()

A single request



Web-server from a mile high: things that matter

1. Network latency/jitter
2. Storage performance
3. Kernel performance
4. Web-server performance

VM/Scheduling from a mile high:

1. Allocate each thread memory

2. Allocate process time on the CPU

3. Swap process to the CPU

4. Swap process from the CPU

Methodology

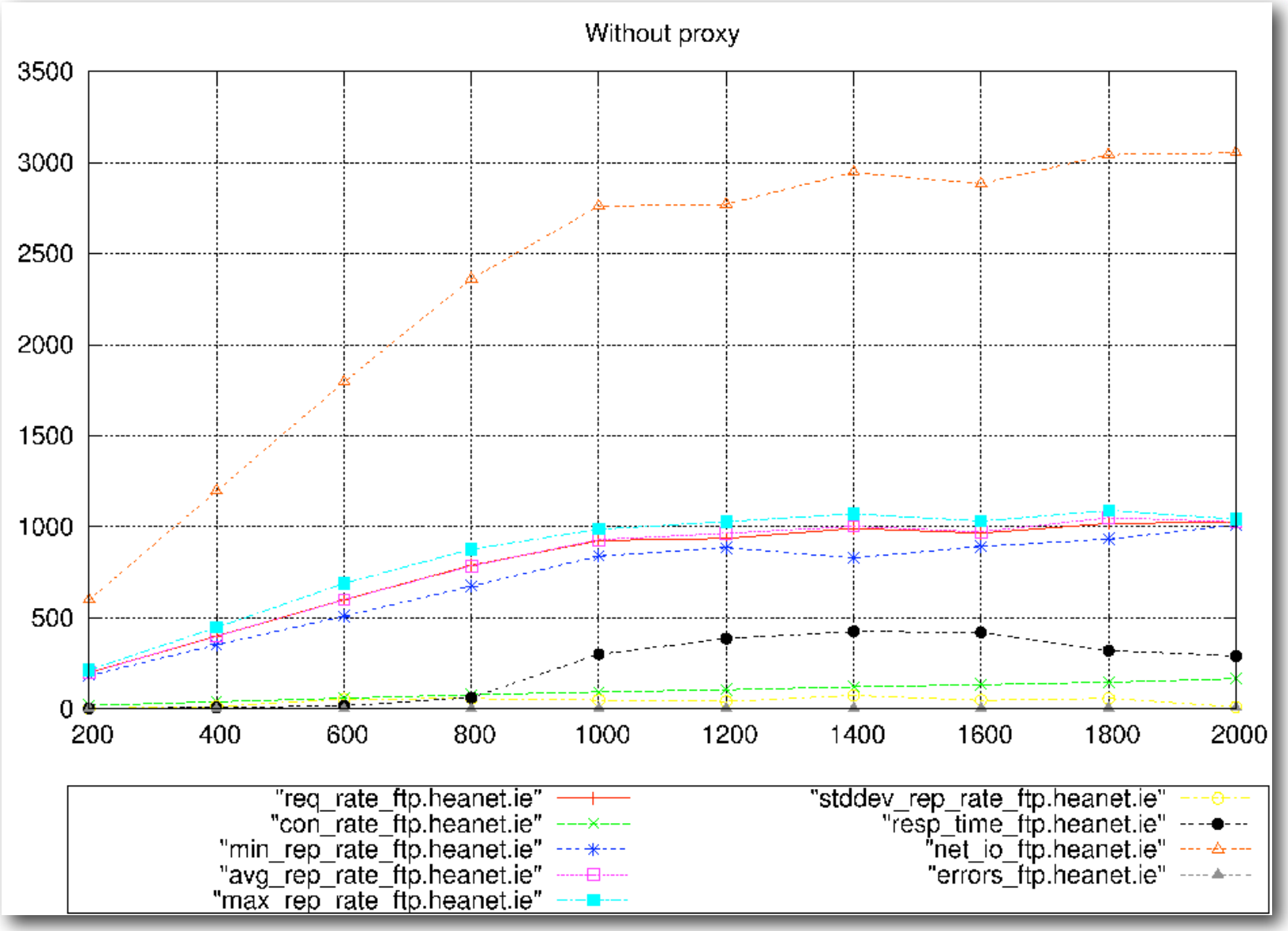
- Research the principles
- Configure, test, benchmark
- Configure, test, benchmark

Benchmarking

- Webserver benchmarking:
  - apachebench, httperf, autobench, siege, SpecWEB
- Always use the same files for benchmarking;
  - /ftp/pub/100.txt
  - /ftp/pub/1000.txt
  - /ftp/pub/10000.txt

Benchmarking

- Ab gives a good overview of webserver performance, can scale well.
- SpecWEB is good, but expensive, especially if you want to publish results.
- httperf + autobench stress-tests and produces useful graphs, to visualise the maximum response rate, error rates, etc.


Without proxy
3500
3000
2500
2000
1500
1000
500
0
200
400
600
800
1000
1200
1400
1600
1800
2000
"req_rate_ftp.heanet.ie"
"con_rate_ftp.heanet.ie"
"min_rep_rate_ftp.heanet.ie"
"avg_rep_rate_ftp.heanet.ie"
"max_rep_rate_ftp.heanet.ie"
"stddev_rep_rate_ftp.heanet.ie"
"resp_time_ftp.heanet.ie"
"net_io_ftp.heanet.ie"
"errors_ftp.heanet.ie"

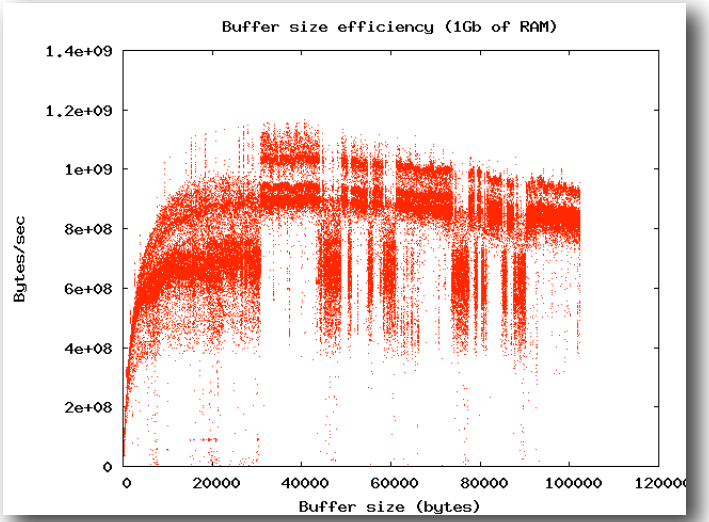
Benchmarking Filesystems and storage

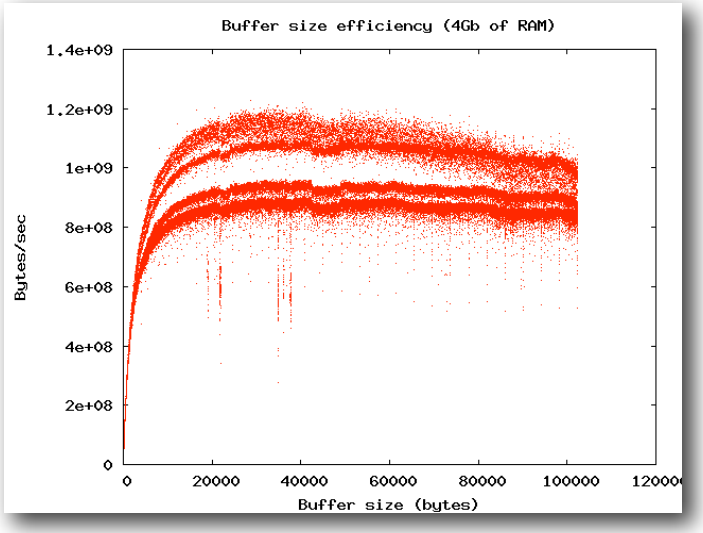
- IOZone, Postmark, Bonnie++, dbench
  - Postmark aimed at simulating mail-spools
  - IOZone is extensive and thorough
  - bonnie++ is simpler, and sufficient for most needs
  - dbench combines disk and TCP benchmarks

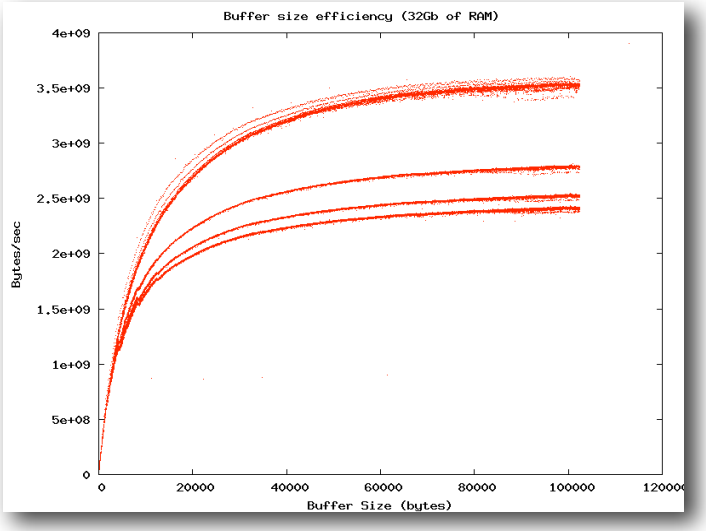
Benchmarking the scheduler and VM

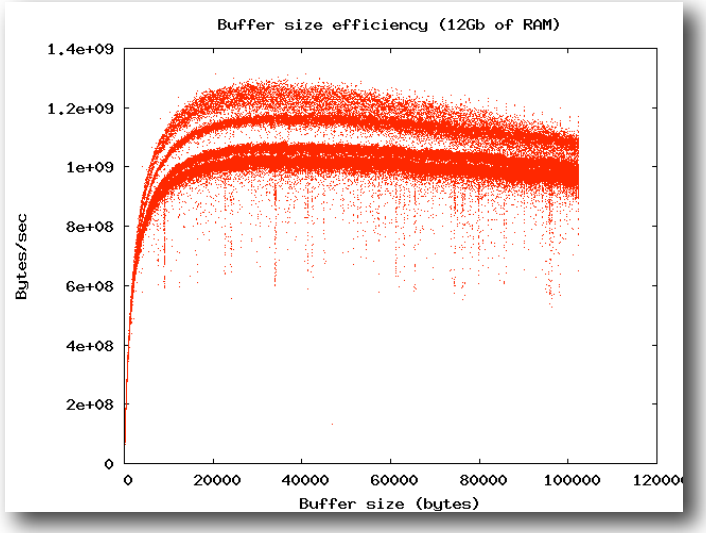
- Very few tools for benchmarking schedulers or VMs. tiobench can be useful.
- Ad-hoc benchmarks usually consist of compiling a kernel
- We use dder.sh

```
#!/bin/sh

STARTNUM="1"
ENDNUM="102400"

# create a 100 MB file
dd bs=1024 count=102400 if=/dev/zero of=local.tmp

# Clear the record
rm -f record

# Find the most efficient size
for size in `seq $STARTNUM $ENDNUM`; do
        dd bs=$size if=local.tmp of=/dev/null                    2>> record
done

# get rid of junk
grep  "transf" record | awk '{ print $7 }' | cut -b 2- | cat -n | \
while read number result ; do
        echo -n $(( $number + $STARTNUM - 1 ))
        echo " " $result
done > record.sane
```


Buffer size efficiency (1Gb of RAM)
Bytes/sec
Buffer size (bytes)
1.4e+09
1.2e+09
1e+09
8e+08
6e+08
4e+08
2e+08
0
0
20000
40000
60000
80000
100000
120000


Buffer size efficiency (4Gb of RAM)
Bytes/sec
Buffer size (bytes)
1.4e+09
1.2e+09
1e+09
8e+08
6e+08
4e+08
2e+08
0
0
20000
40000
60000
80000
100000
120000


Buffer size efficiency (32Gb of RAM)
Bytes/sec
Buffer Size (bytes)
4e+09
3.5e+09
3e+09
2.5e+09
2e+09
1.5e+09
1e+09
5e+08
0
0
20000
40000
60000
80000
100000
120000


Buffer size efficiency (12Gb of RAM)
Bytes/sec
Buffer size (bytes)
1.4e+09
1.2e+09
1e+09
8e+08
6e+08
4e+08
2e+08
0
0
20000
40000
60000
80000
100000
120000

Tuning Apache

- Benchmark the MPMs
  - worker and event currently on top
- Static or DSO build for modules
  - miniscule difference
- AllowOverride none / EnableSendfile / EnableMMap

mod_cache

- Experimental in 2.0, but great in 2.2
- Not just for proxies, allows web-servers to cache files as they are requested
- Many reads from a slow file-system can be avoided

Tuning the VM

- Compile Apache with -Os
  - Doesn't speed up code much, but helps the VM
- Add more RAM

Tuning the Operating System

- Choose a kernel
  - Linux 2.6 is much better than 2.4
  - Solaris Express (nevada) better than Solaris 10
- Tune the filesystem
  - always mount with noatime
  - XFS: use logbufs=8, ihashsize=65567
  - EXT3: set blocksize to 4096, use dir_index build option

Sysctl

- vm/min_free_kybtes
- vm/lower_zone_protection
- vm/page-cluster
- vm/swappiness
- vm/vm_vfs_scan_ratio
- fs/file-max

Sysctl

- net/ipv4/tcp_rfc1337
- net/ipv4/tcp_syncookies
- net/ipv4/tcp_keepalive_time
- net/ipv4/tcp_max_orphans
- sys/net/core/wmem_default
- sys/net/core/wmem_max

System Design

- Lots of Memory
- Bounce buffering and PAE to be avoided, otherwise lots of CPU
- Fast (15k RPM) SCSI disks for caching

Architectures

- Xeons
- Itanium
- Niagara
- Opteron

Questions

?