



OpenAFS Best Practices Workshop  
Stanford, California  
Mar 24-26, 2004

# OpenAFS Client Performance Analysis

Kris Van Hees  
Sine Nomine Associates

[kvanhees@sinenomine.net](mailto:kvanhees@sinenomine.net)

# Problems

- ▶ Reading from servers  $\ll$  wire speed
- ▶ Writing to servers  $\ll$  wire speed
- ▶ Writing to cache seems relatively slow
- ▶ The code involved is very complex
- ▶ Tuning is quite difficult
  
- ▶ Anything we're forgetting?

# Proposal

- ▶ Focus on:
  - Cache manager
  - Client RX communication layer
  - Interaction with kernel
- ▶ Detailed analysis of various components
- ▶ Controlled testing of throughput under varying loads
- ▶ Limited to Linux (for now)

# Assumptions

- ▶ Server side is not a bottleneck
  - Caveat: Server RX communication?
- ▶ 100Base-TX ought to be sufficient
- ▶ No “pre-test” tweaking of code
- ▶ Linux kernel influence is ignored

# Goals

- ▶ Identify bottlenecks in client code
- ▶ Provide scientific data to support findings
- ▶ Offer recommendations for improvements to the client code
- ▶ Determine whether other factors are to be considered

# Procedure

## ▶ Iterative approach

1. Add instrumentation code to OpenAFS
2. Perform tests
3. Analyze both internal and external measurements
4. Refine instrumentation based on analysis
5. Iterate to step 2

# Measurements

- ▶ Overall throughput
- ▶ User level throughput
- ▶ Kernel level throughput
- ▶ Network utilization
- ▶ RX protocol timing
- ▶ Cache manager timing
- ▶ System load

# Load generation

## ▶ Raw write load generation

- Single client

- | File Count | File Size (MB) |
|------------|----------------|
| 1900       | 1              |
| 950        | 2              |
| 475        | 4              |
| 237        | 8              |
| 118        | 16             |
| 59         | 32             |
| 29         | 64             |
| 14         | 128            |
| 7          | 256            |
| 3          | 512            |
| 1          | 1024           |



# Load generation (cont...)

## ▶ Raw read load generation

- Multiple clients, operating in parallel

- | File Count | File Size (MB) |
|------------|----------------|
| 100        | 1              |
| 100        | 2              |
| 100        | 4              |
| 100        | 8              |
| 100        | 16             |
| 59         | 32             |
| 29         | 64             |
| 14         | 128            |
| 7          | 256            |
| 3          | 512            |
| 1          | 1024           |

# Load generation (cont...)

- ▶ Combined multi-file load generation
  - Multiple clients, operating in parallel

- |            | Test 1         |            | Test 2         |  |
|------------|----------------|------------|----------------|--|
| File Count | File Size (MB) | File Count | File Size (MB) |  |
| 100        | 1              | 100        | 1              |  |
| 100        | 2              | 100        | 2              |  |
| 100        | 4              | 100        | 4              |  |
| 100        | 8              | 100        | 8              |  |
| 100        | 16             |            |                |  |

# Future directions

- ▶ Performance analysis has been submitted as an SBIR project with the DOE
- ▶ Comparison with independent testing by production users
- ▶ Comparison with testing on other platforms
- ▶ Algorithmic changes
- ▶ Server side optimizations
- ▶ ...

- Kris Van Hees
- Sine Nomine Associates
  - 43596 Blacksmith Sq
  - Ashburn, VA 20147
  - (703) 723-6673
- [kvanhees@sinenomine.net](mailto:kvanhees@sinenomine.net)