

Performance evaluation on the SCC Cluster

The following numbers were gathered on the SCC Cluster in Karlsruhe: <http://www.scc.kit.edu/en/index.php>

Caveat: At the creation of the benchmarks there were a lot of VMs running on those 14 hosts (62) which have of course an impact on the performance of Fedora 4 since all machines share the I/O channels of one host. The hosts are connected via a 1GB/s network, and I think the two virtual partitions per VM are also mounted via this network. I am currently in contact with the folks at SCC to maybe get exclusive access to a couple of physical nodes, so that other VMs do not slow down Fedora 4 performance.

When comparing D and E (with and without indexing), there should be an increase in performance, when turning indexing off. Since this is not the case I'm guessing that the I/O bottleneck is hit even earlier (replication over the network?) so that indexing does not slow down the ingest process at all.

Node network I/O performance

The physical hosts have a 1gb/s network connection but I measured the network performance to be ~ 10MB/s when pushing one file from one VM to another VM over the network. This is probably due to the fact that multiple VMs share the I/O channel of one physical host

Node hdd performance

```
ubuntu@ ubuntu:/data$ sync;time sudo bash -c "(dd if=/dev/zero of=bf bs=8k count=500000; sync)"
```

```
500000+0 records in
500000+0 records out
4096000000 bytes (4.1 GB) copied, 105.135 s, 39.0 MB/s
```

```
real 2m34.033s
user 0m0.060s
sys 0m5.590s
```

CPU

Following is the output of 'cat /proc/cpuinfo' on one VM

```
vendor_id : GenuineIntel
cpu family : 6
model : 6
model name : QEMU Virtual CPU version 0.9.1
stepping : 3
cpu MHz : 2266.804
cache size : 32 KB
fpu : yes
fpu_exception : yes
cpuid level : 4
wp : yes
flags : fpu de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pse36 clflush mmx fxsr sse sse2 syscall nx lm rep_good nopl pni hypervisor
bogomips : 4533.60
clflush size : 64
cache_alignment : 64
address sizes : 40 bits physical, 48 bits virtual
power management:
```

```
processor : 1
vendor_id : GenuineIntel
cpu family : 6
model : 6
model name : QEMU Virtual CPU version 0.9.1
stepping : 3
cpu MHz : 2266.804
cache size : 32 KB
fpu : yes
fpu_exception : yes
cpuid level : 4
wp : yes
flags : fpu de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pse36 clflush mmx fxsr sse sse2 syscall nx lm rep_good nopl pni hypervisor
bogomips : 4533.60
clflush size : 64
cache_alignment : 64
address sizes : 40 bits physical, 48 bits virtual
power management:
```

Load balancing

Load balancing is done by using an [apache](#) server with `mod_jk` enabled on a dedicated VM and a `jk_workers.properties` file which has the individual nodes configured as `mod_jk` workers. This results in a simple round-robin load balancing mechanism.

The `jk_workers.properties` file is currently being generated via a shell script:

<https://github.com/futures/scc-cluster-install/blob/master/fedora-node.sh#L44>

Example:

To balance between 7 nodes the `jk_workers.properties` file could look like this:

<https://gist.github.com/fasseg/7138008>

Results

Test Utility

BenchTool: <https://github.com/futures/benchtool>

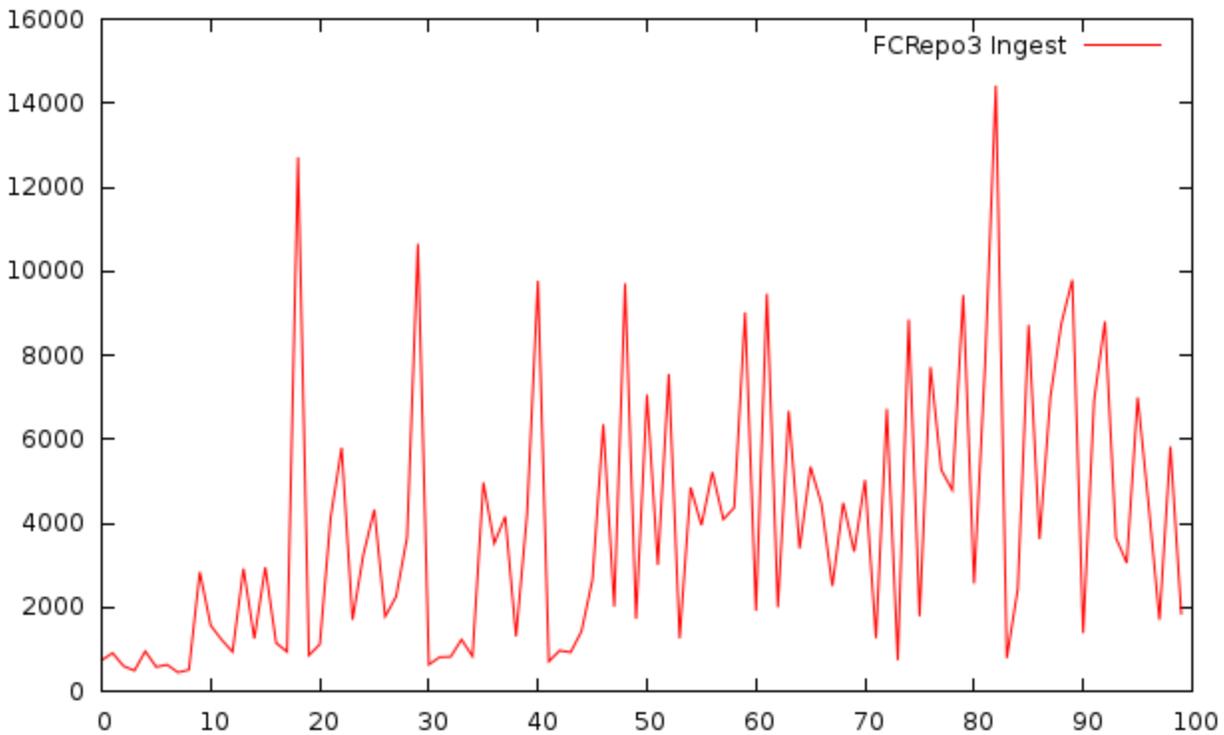
A. Ingest bench using fcrepo3

Created 100 objects with one datastream of 50mb size

Size: $100 * 52428800$ bytes = 5gb

Duration: 389034 ms

Throughput: 12.85 mb/s



B. Ingest bench using single node fcrepo4 with single thread and clustered config

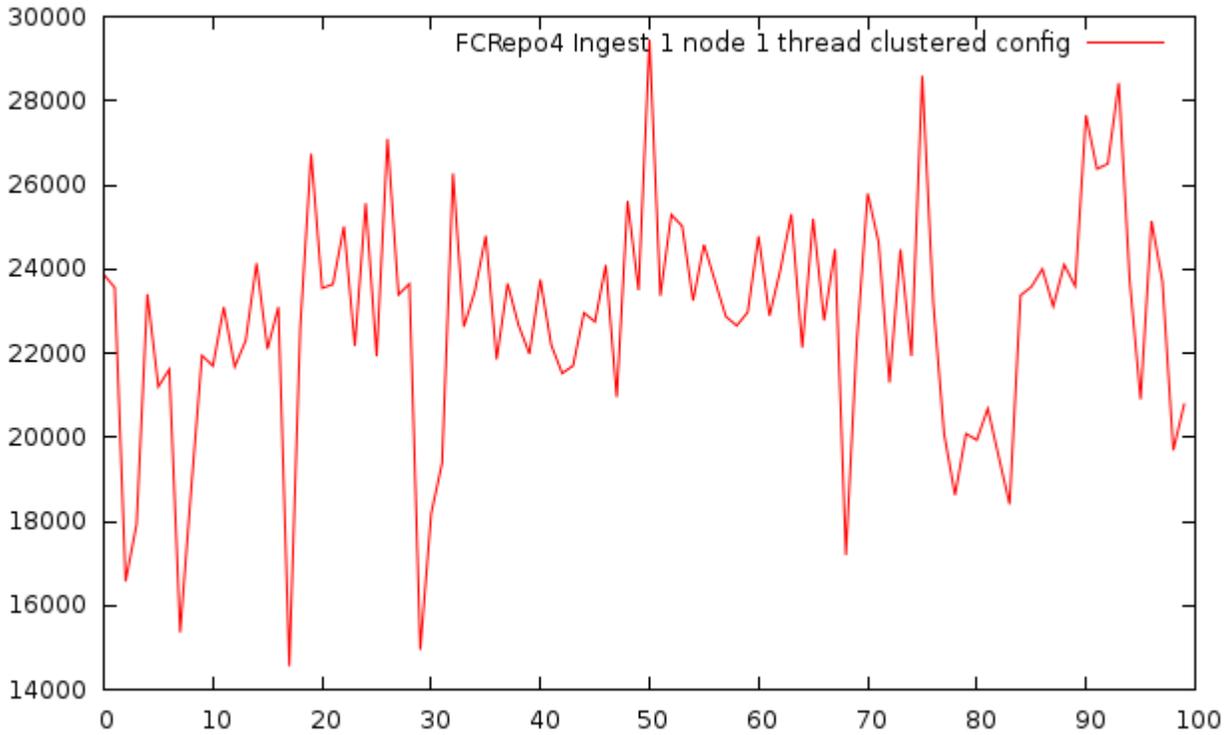
Created 100 objects with one datastream of 50mb size

Size: 100 * 52428800 bytes = 5gb
Duration: 2310171 ms
Throughput: 2.1 mb/s

[Infinispan configuration](#)

[Repository configuration](#)

[JGroups configuration](#)



C. Ingest bench using single node fcrepo4 with single thread and minimal config

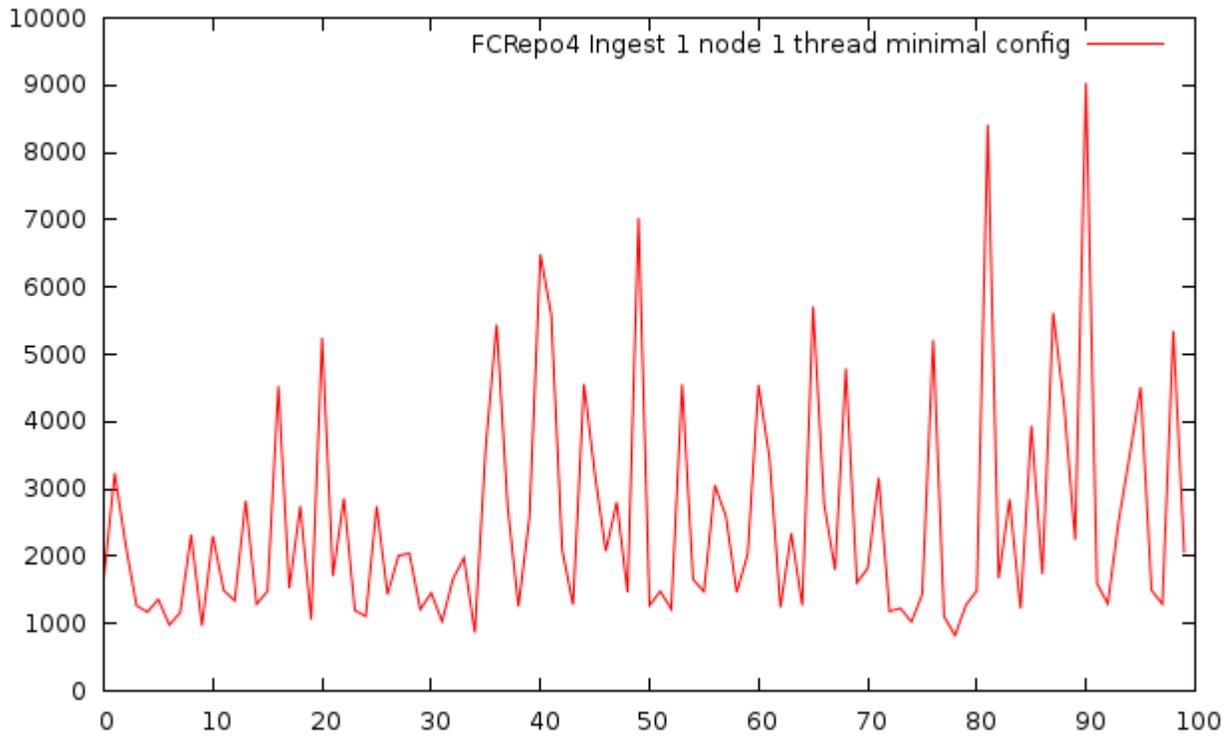
Created 100 objects with one datastream of 50mb size

Size: 100 * 52428800 bytes = 5gb
Duration: 281301 ms
Throughput: 17.77 mb/s

[Infinispan configuration](#)

[Repository configuration](#)

[JGroups configuration](#)



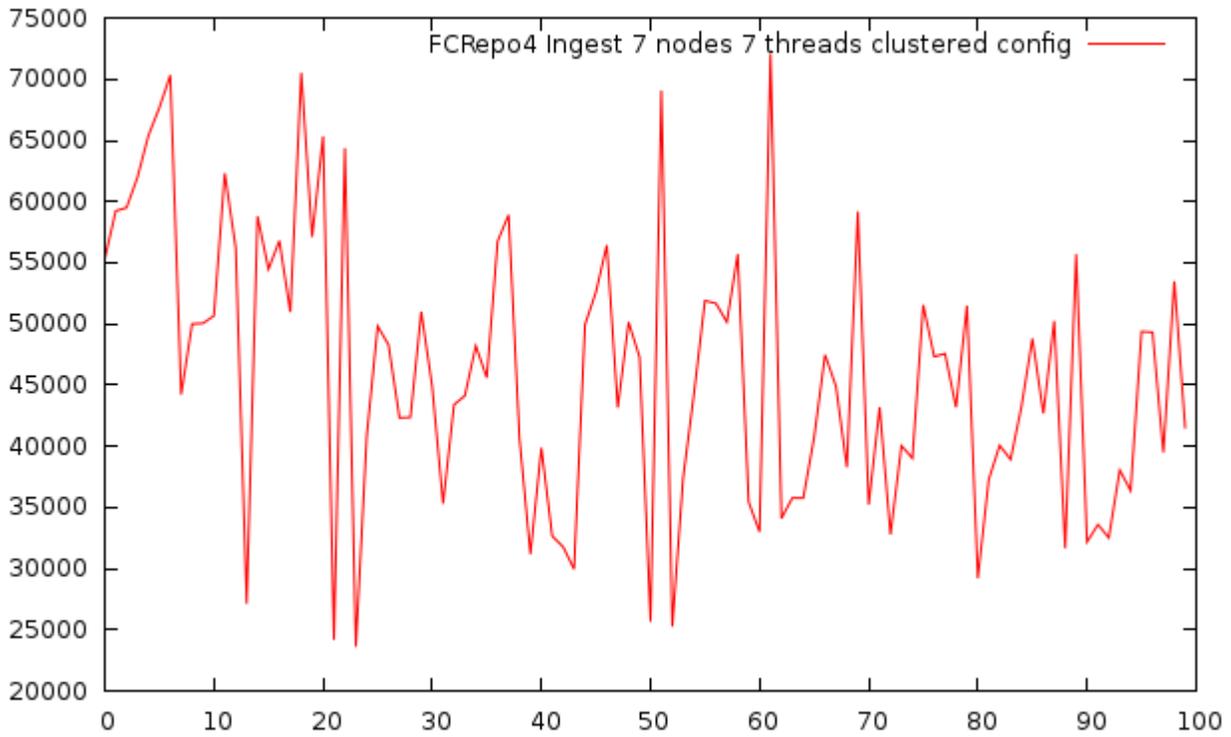
D. Ingest bench using seven nodes fcrepo4 with seven threads and clustered config
Created 100 objects with one datastream of 50mb size

Size: 100 * 52428800 bytes = 5gb
Duration: 676016 ms
Throughput: 7.4 mb/s

[Infinispan configuration](#)

[Repository configuration](#)

[JGroups configuration](#)



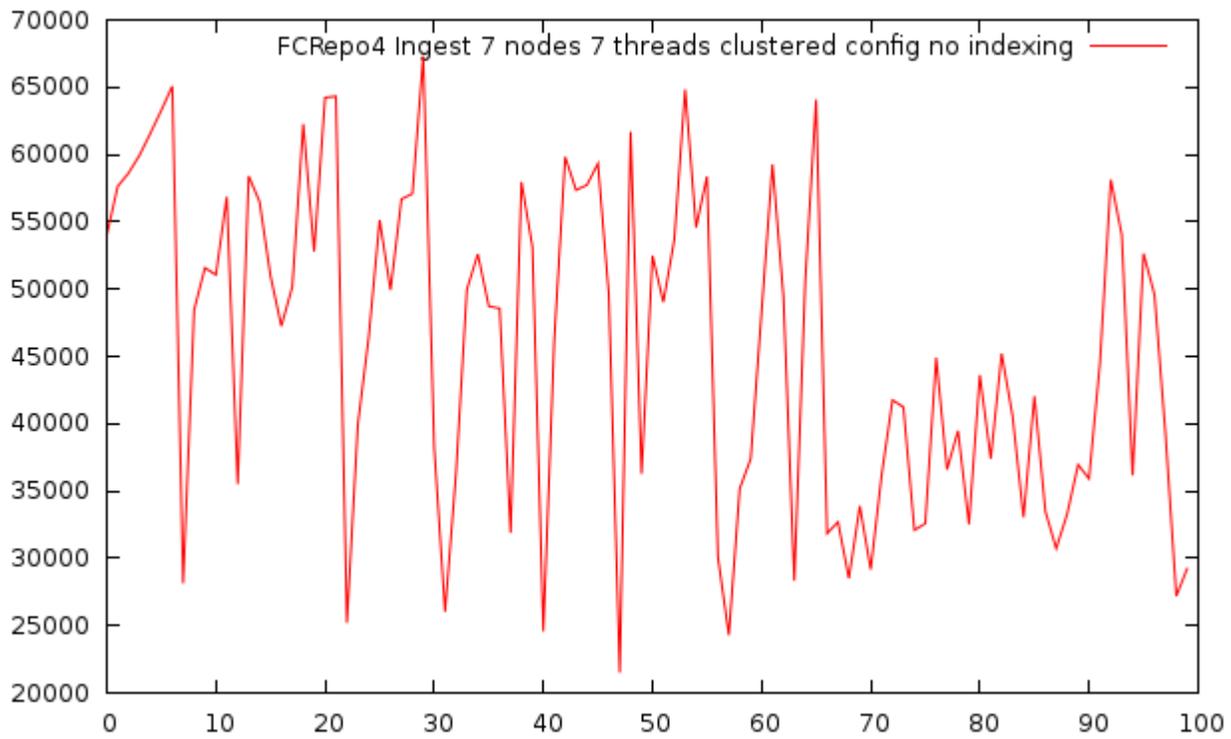
E. Ingest bench using seven nodes fcrepo4 with seven threads and clustered config, no indexing
Created 100 objects with one datastream of 50mb size

Size: 100 * 52428800 bytes = 5gb
Duration: 673589 ms
Throughput: 7.4 mb/s

[Infinispan configuration](#)

[Repository configuration](#)

[JGroups configuration](#)



F. Ingest bench using seven nodes fcrepo4 with eleven threads and clustered config, no indexing
Created 100 objects with one datastream of 50mb size

Size: 100 * 52428800 bytes = 5gb
Duration: 628219 ms
Throughput: 8.0 mb/s