

Improvement for performance of memcached



Kazuaki MASUDA (phi) — ARCH, Jun MURAI Lab, Keio University, Japan

Background

Memcached

- A distributed memory cache server
- Reducing the number of communications to the DB and prevents overload by caching the results of the DB transaction
- Using it for automatic updating web application to speed up

Problem & Goal

Memcached is an in-memory database server.



The processing time for protocol stack is a bottleneck in the whole.



Reducing that processing time and improving performance of memcached.

Approach

Improve memcached's performance by using the packet I/O framework named netmap, handling protocols on the application side and reducing processing time.

netmap

- a framework for fast packet I/O from userspace
- linear, fixed size packet buffers that are preallocated.
 - removing the cost of per-packet allocations and deallocations.
- Removal of data-copy costs by granting applications direct, protected access to the packet buffers.

Implementation

- Using UDP packets.
- Creating protocol header and attach to netmap's raw packet for UDP communication with netmap.
- Create function by wrapping systemcall "send", "recv" and replacing them.

Related work

Warpcore

NetApp, Inc.

- <https://github.com/NTAP/warpcore>
- Simple UDP/IP protocol stack for netmap framework.
- It prioritizes performance over features, and full standards compliance.

Evaluation

- Comparing performance with the normal memcached.
- Using mcb, a memcached benchmark program
 - <https://github.com/s-hironobu/mcb/>
- Comparing with memcached using warpcore.

Schedule

