

Varnish

high-performance HTTP accelerator

Edward Middleton

Introduction

Modern design - doesn't churn swap like squid

Uses a flexible DSL for configuration

compiles DSL to native library

Supports

load balancing

some ESI (includes, etc.)

purging specific cache items

Support backend health check

My objectives

Replace filesystem caching in typo blog

Remove all unnecessary access to backend

Reduce page latency

Increase throughput

Increase scalability

Find a reusable way to do this for Ruby on Rails

Finish before lightening talk!!

My setup

Celeron 1.6 Ghz (2GB RAM, 100mb fiber)

Web facing varnish proxy

Nginx load balancing

thin application servers

MySQL database



The Plan

Replace existing Page Cache

Write to cache sets Cache-Control header.

Prime varnish with first read.

Expire causes PURGE request to varnish.

Make browser check for update before displaying cached page.

How far I got

Remove page cache causes PURGE request.

Replaced the page cache