Scaling SourceForge with MongoDB

Nosh Petigara
@noshinosh
nosh@10gen.com

Rick Copeland
@rick446
rick@geek.net
One-Size-Fits-All No Longer

- **RDBMS**
  - (Oracle, MySQL)

- **New gen. OLAP**
  - (vertica, aster, greenplum)

- **Non-relational operational stores**
  - (“NoSQL”)
The Great Divide

‘NoSQL’ Sweet spot: Fast, Flexible, Scalable?
NoSQL?

Lots of differences

  Consistency model (CP, AP, etc)
  Scale out model
  Query languages

What’s the same?

  No JOINs
  No complex transactions
Anatomy of a Document

{  
_id: "A4304",
author: "nosh",
date: 22/6/2010,
title: "OSCON 2010",
text: "Portland was great…",
tags: ["conference", "opensource"],
comments: [{author: "mike",
  date: 22/6/2010,
  txt: "Did you see the…",
  votes: 7},....]
}

Documents -> Collections -> Databases
Queries

db.posts.find({author: "nosh",
               tags: "conference"})

$gt, $lt, $gte, $lte, $ne, $all, $in, $nin, count, limit, skip, group

Where is my join??!!
Updates

db.posts.update({_id:A4304},
   {title: "OSCON 2010 (updated)",
    text: "Free coffee at the MongoDB booth"},
   ${addToSet: {tags: "mongodb"}})

$set, $unset

$push, $pop, $pull, $addToSet

$inc

Where is my transaction??
The Scale Story

No JOINs + No complex transactions = Horizontal Scalability!

- Single master mode
- Auto-sharded mode

‘NoSQL’ Sweetspot: Fast, Flexible, Scalable
Links

http://mongodb.org
http://10gen.com
http://github.com/mongodb
http://slideshare.net/mongodb
@mongodb
nosh@10gen.com

Visit us at our booth!

MongoDB: Fast, Flexible, Scalable
SF.net “BlackOps”: FossFor.us

User Editable!

Web 2.0! (ish)

Not Ugly!
Moving to NoSQL

- FossFor.us used CouchDB (NoSQL)
- “Just adding new fields was trivial, and was happening all the time” – Mark Ramm
- Scaling up to the level of SF.net needs research
  - CouchDB
  - MongoDB
  - Tokyo Cabinet/Tyrant
  - Cassandra... and others
What we were looking for

- Performance – how does a single node perform?
- Scalability – needs to support simple replication
- Ability to handle complex data and queries
- Ease of development
Rewriting “Consume”

- Most traffic on SF.net hits 3 types of pages:
  - Project Summary
  - File Browser
  - Download
- Pages are read-mostly, with infrequent updates from the “Develop” side of sf.net
- Original goal is 1 MongoDB document per project
  - Later split release data because some projects have lots of releases
- Periodic updates via RSS and AMQP from “Develop”
Deployment Architecture (revised)

- Load Balancer / Proxy
- Apache mod_wsgi / TG 2.0
- Scalability is good
- Single-node performance is good, too
- Master DB Server
  - MongoDB Master
- Gobble Server
- Develop

Geeknet and 10gen, page 15
Benefits and Pitfalls

- Performance, performance, performance – comfortably handles 75+ dynamic page requests per second per server
- Schemaless server allows fast schema evolution in development, making many migrations unnecessary
- Replication is easy, making scalability easy and backups easy
  - Keep a “backup slave” running
  - Kill backup slave, copy off database, bring back up the slave
  - Automatic re-sync with master
- Pitfall: returning more data than you need
  - Good news: you'll max out your network bandwidth before you max out your CPU usage on the MongoDB server
  - Bad news: you'll max out your network bandwidth
- Pitfall: bad indexing
New Stuff: Downloads

- Allow non-sf.net projects to use SourceForge mirror network
- Nice statistics and reporting calculated in Hadoop and stored/served from MongoDB
- Same deployment architecture as Consume
New Stuff: SF.net 2.0 beta

- Rewrite developer tools with new architecture
- Wiki, Issue Tracker, Discussions, Git, SVN available today (with APIs), more to come
- Single MongoDB master server, one database per project
  - Tricky to get right!
- Release early & often
Open Source and Future Ideas

- Ming* of MongoDB – a Python “Object Document Mapper” - enforce a schema and migrations in your application code
- More APIs allowing better project data access
- Project data portability – replicate your Project database from sf.net
- New tools and applications for developers
- Look for new OSS projects coming out of SF.net....

* The name Ming is in conflict with a Flash framework of the same name; we might be changing the name soon. It should, however, always be available under http://sourceforge.net/projects/merciless
Questions?
Contact

Nosh Petigara
@noshinosh
nosh@10gen.com

Rick Copeland
@rick446
rick@geek.net