General Ops Notes

Getting a single instance started

Need a directory for data files:

```
mkdir /data/db
```

Start up mongod:

```
mongod --dbpath /data/db
```

Things to add in production. Use mongod --help for all options:

```
--logpath, --logappend, --fork, --rest, --port, ...
```

Use a config file:

```
dbpath = /data/db
logpath = /path/to/logfile
logappend = true
fork = true
rest = true
port = 27018
```

Then use from the command line:

```
mongod --config /path/to/config/file
```

Connect using the mongo shell:

```
mongo
```

Generate some data:

```
use training

for(i=0; i<10000; i++) {
  ['quiz', 'essay', 'exam'].forEach(function(name) {
     var score = Math.floor(Math.random() * 50) + 50;
     db.scores.insert({student: i, name: name, score: score});
  });
}
```

Monitoring

Important tools:

```
mongostat (--discover)
iostat -x 2
http://localhost:<1000 greater than --port>
```

```
mms.10gen.com
```

Getting stats from the shell:

```
db.serverStatus()
db.stats()
db.<collection name>.stats()

# Do this in another shell to generate load
db.foo.drop()
for (var i=0; i<10000; i++){
    db.foo.insert({_id: i});
    for (var j=0; j< 10; j++){
        db.foo.findOne({_id: (i-j)});
    }
}
```

See what the server is currently doing:

```
db.currentOp()
db.killOp()

# Start a db.repairDatabase(), find it in currentOp, then kill with killOp.
```

Plugins for external tools:

```
Munin, Nagios, Cacti, Ganglia
```

Backup

For backup/restore on a live system:

```
mongodump -d <database> -c <collection> ...
mongorestore -d <database> -c <collection> ...
```

You can also copy/rsync/snapshot the data files:

```
# Make sure you lock the db first
db.fsyncLock()
db.fsyncUnlock()

# On MongoDB pre 2.0
db.runCommand({fsync: 1, lock: 1})
db.$cmd.sys.unlock.findOne()
```

Usually you want to do backups from a slave/secondary.