Highly available clustered apps on Amazon Web Services

Or how AWS saves us money

Tomasz Szymański @szimano

softwaremill

Who am I?

- SoftwareMill
  - Extraordinary Software as as Standard
  - http://softwaremill.com
- Tomasz Szymański:
  - Warszawa-JUG leader
  - http://szimano.org

Agenda

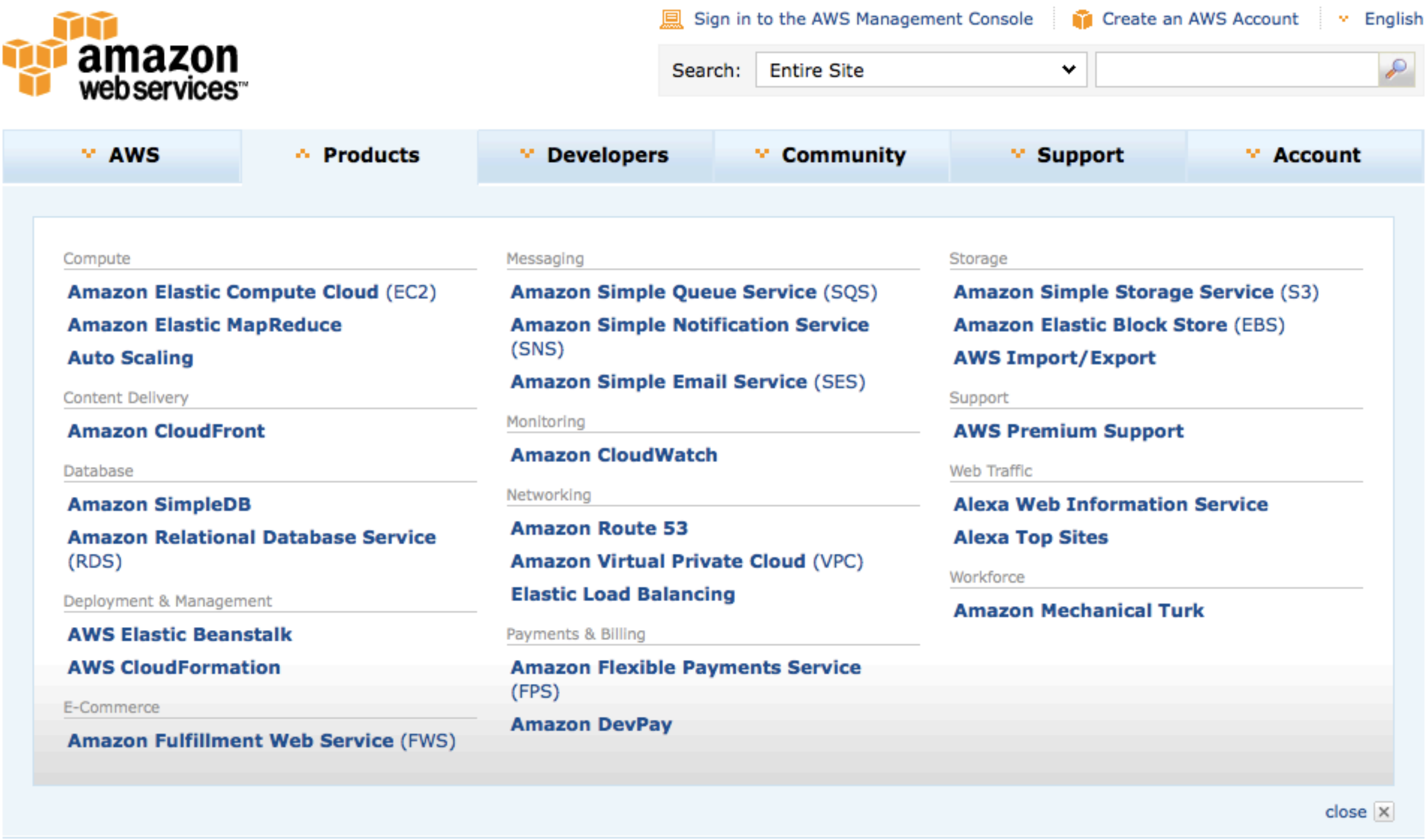
- Short Intro: History of AWS, what is it ?
- Demo1: app without the cloud
- Demo2: app with the cloud

Short history

- .com boom – Amazon builds huge server centers in USA
- The bubble bursts – Amazon’s need for the servers is much smaller then they originally though
- Amazon decides to sell the computing power „per hour”
- July 2002 - Amazon Web Services starts

AWS – what is it?

- Set of different services:
  - EC2 – Elastic Compute Cloud
  - EBS – Elastic Block Store
  - S3 – Simple Storage Service
  - SQS – Simple Queue Service
  - SNS – Simple Notification Service
  - ELB – Elastic Load Balancing
  - SimpleDB
  - RDS – Relational Database Service
- … and much much more



Search: Entire Site

Compute

- Amazon Elastic Compute Cloud (EC2)
- Amazon Elastic MapReduce
- Auto Scaling

Content Delivery

- Amazon CloudFront

Database

- Amazon SimpleDB
- Amazon Relational Database Service (RDS)

Deployment & Management

- AWS Elastic Beanstalk
- AWS CloudFormation

E-Commerce

- Amazon Fulfillment Web Service (FWS)

Messaging

- Amazon Simple Queue Service (SQS)
- Amazon Simple Notification Service (SNS)
- Amazon Simple Email Service (SES)

Monitoring

- Amazon CloudWatch

Networking

- Amazon Route 53
- Amazon Virtual Private Cloud (VPC)
- Elastic Load Balancing

Payments & Billing

- Amazon Flexible Payments Service (FPS)
- Amazon DevPay

Storage

- Amazon Simple Storage Service (S3)
- Amazon Elastic Block Store (EBS)
- AWS Import/Export

Support

- AWS Premium Support

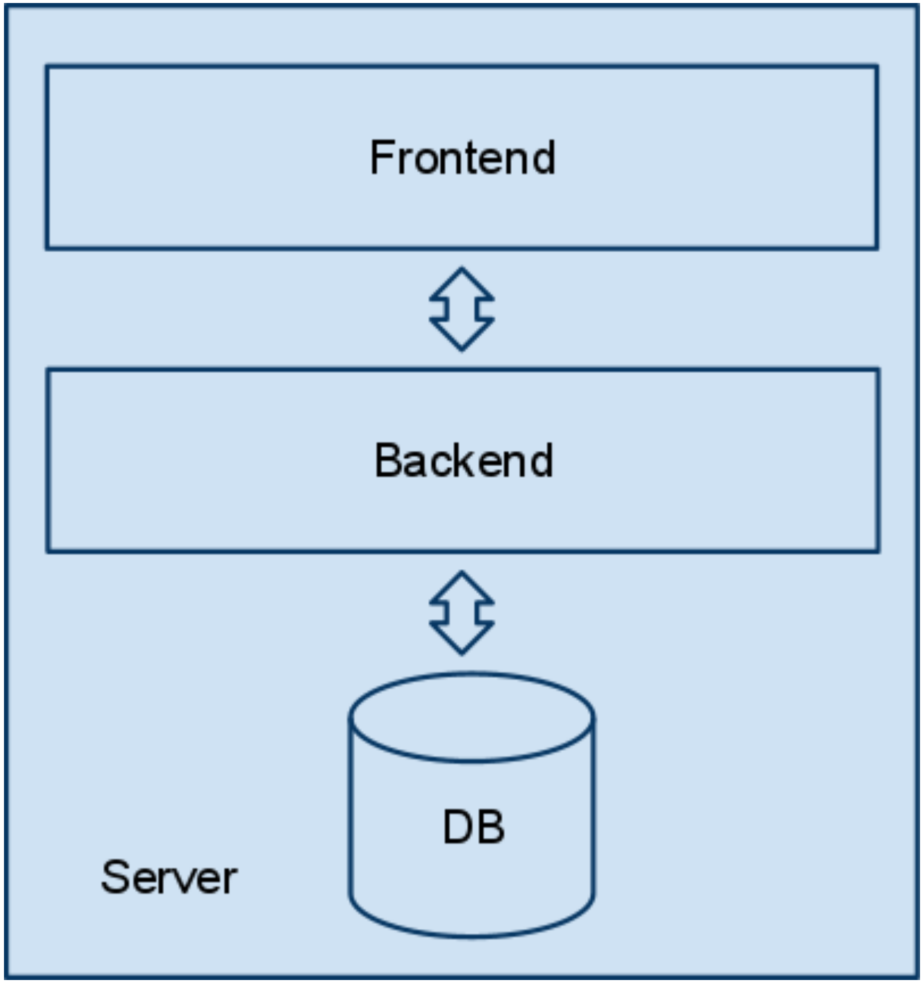
Web Traffic

- Alexa Web Information Service
- Alexa Top Sites

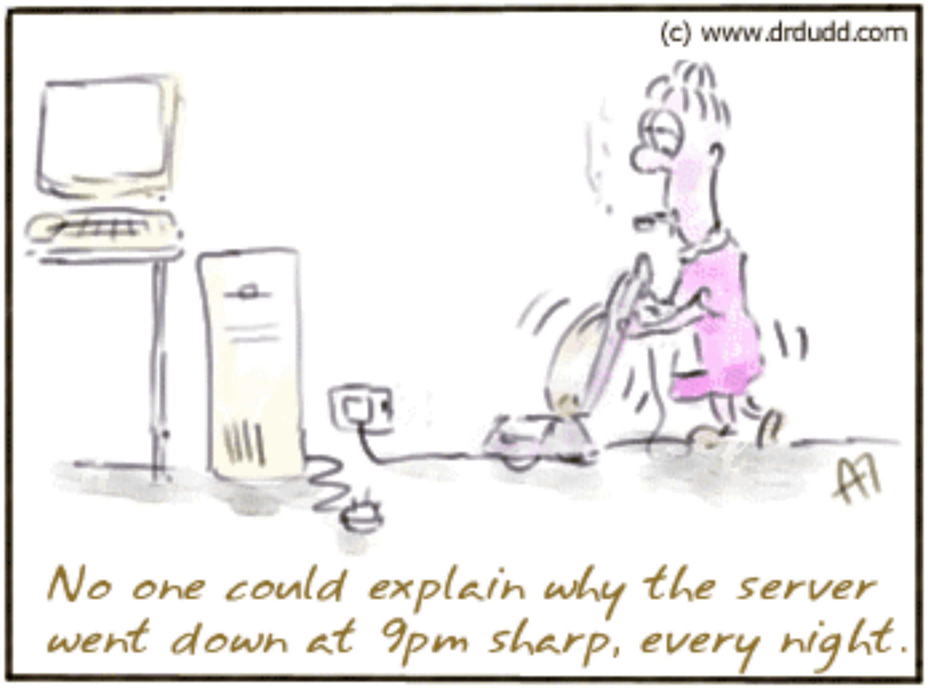
Workforce

- Amazon Mechanical Turk

Our application



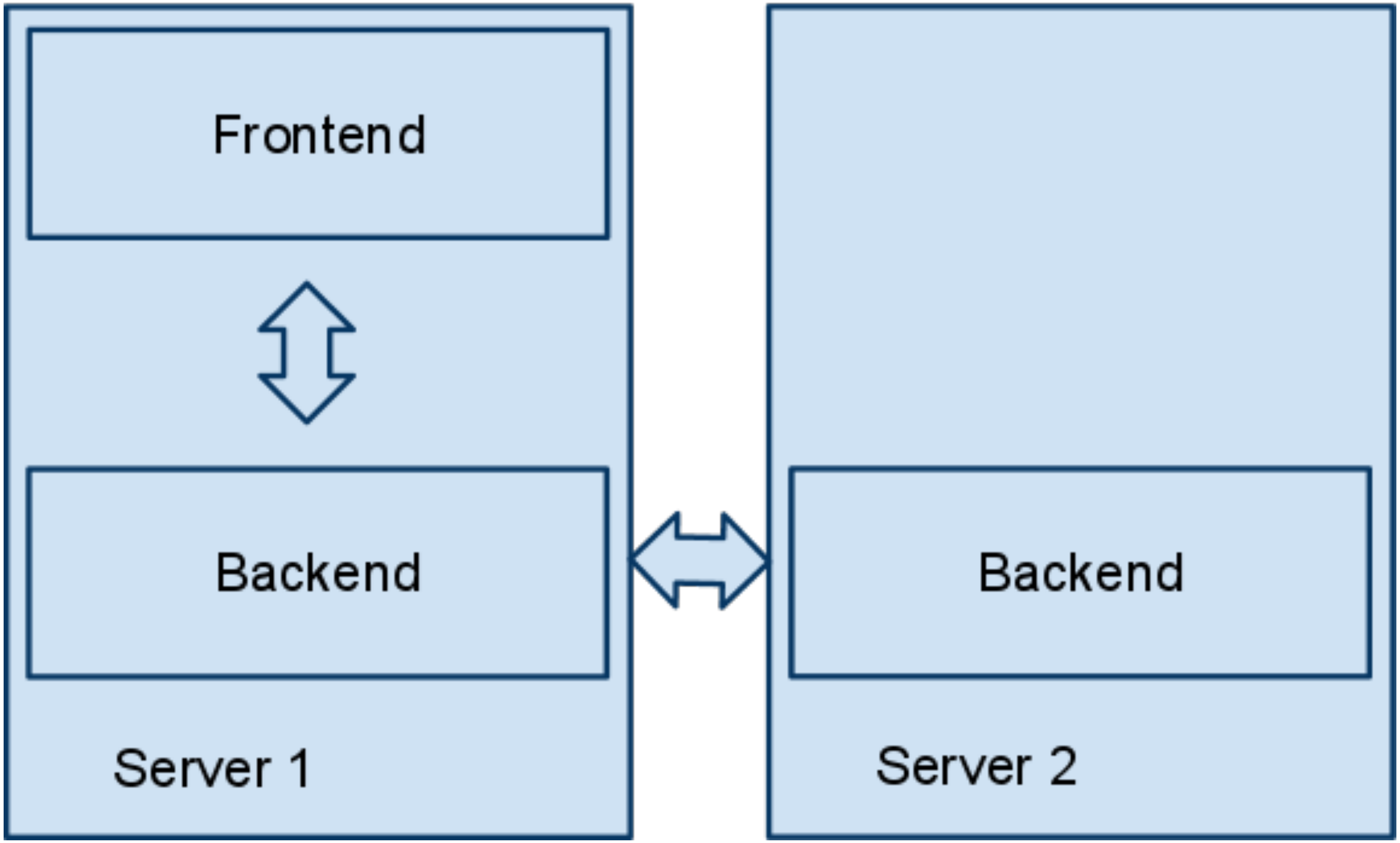
But...


(c) www.drdudd.com
AA
No one could explain why the server went down at 9pm sharp, every night.

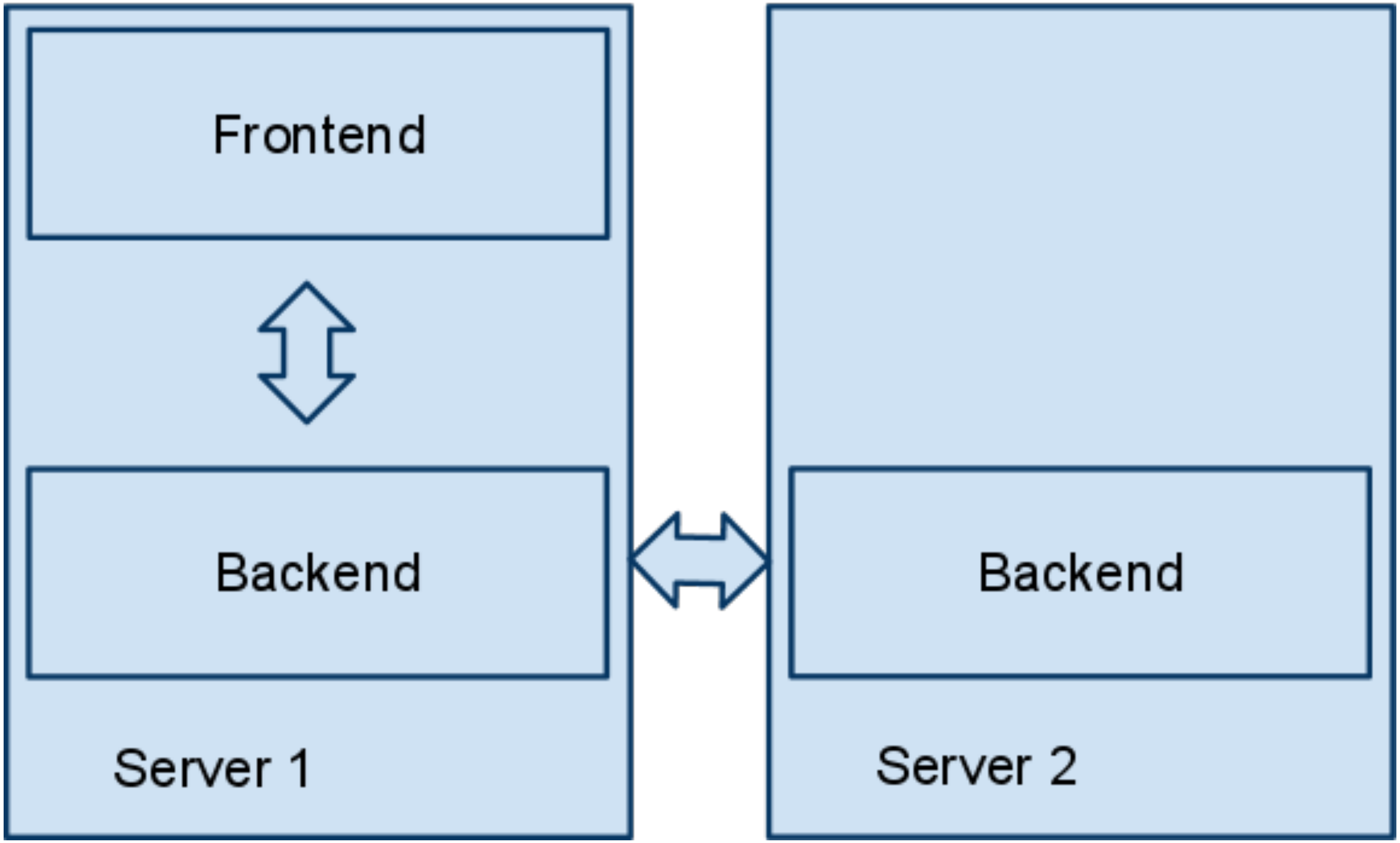
What do we want?

- Scalability
- High availability

Another server

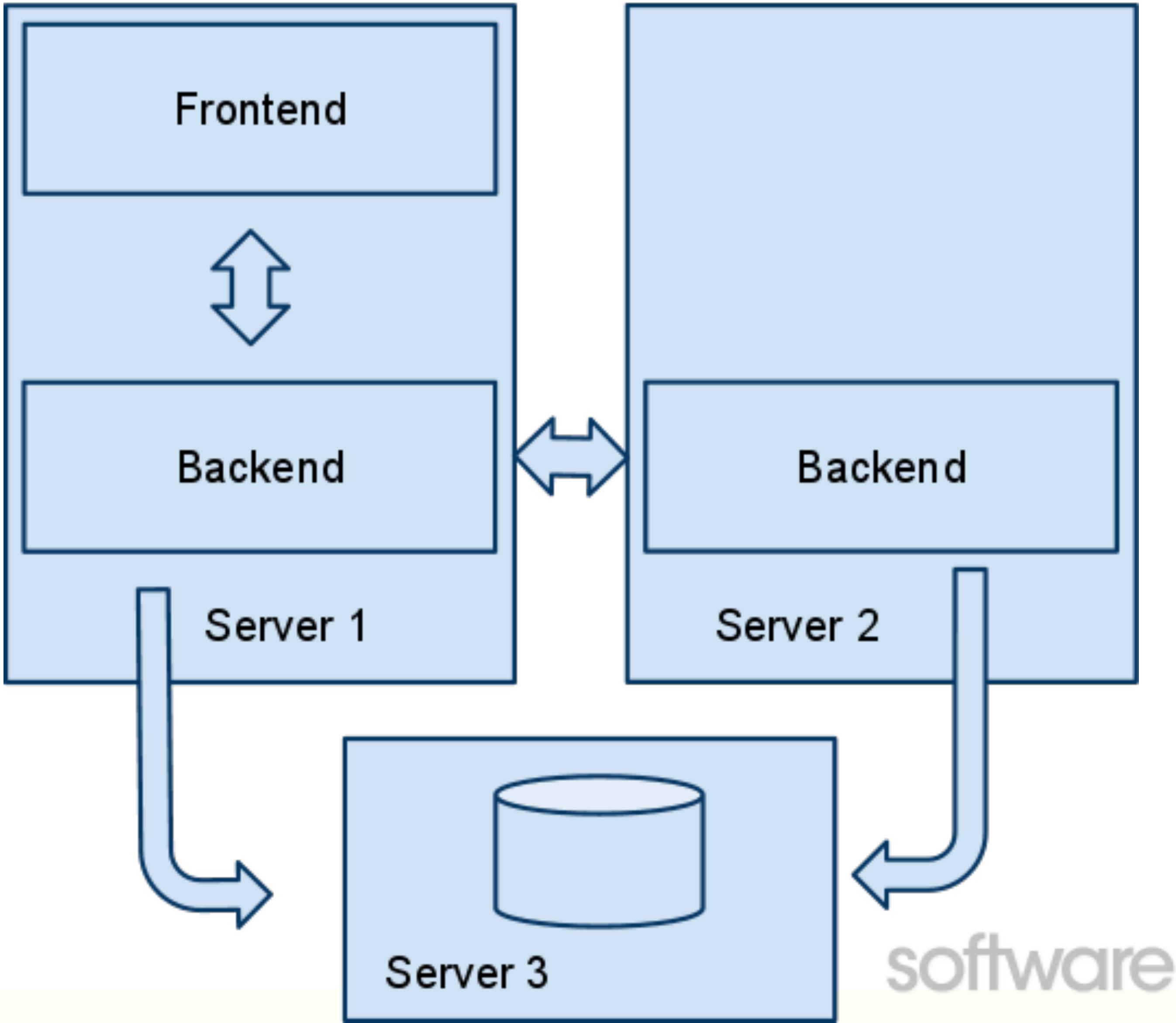


Another server



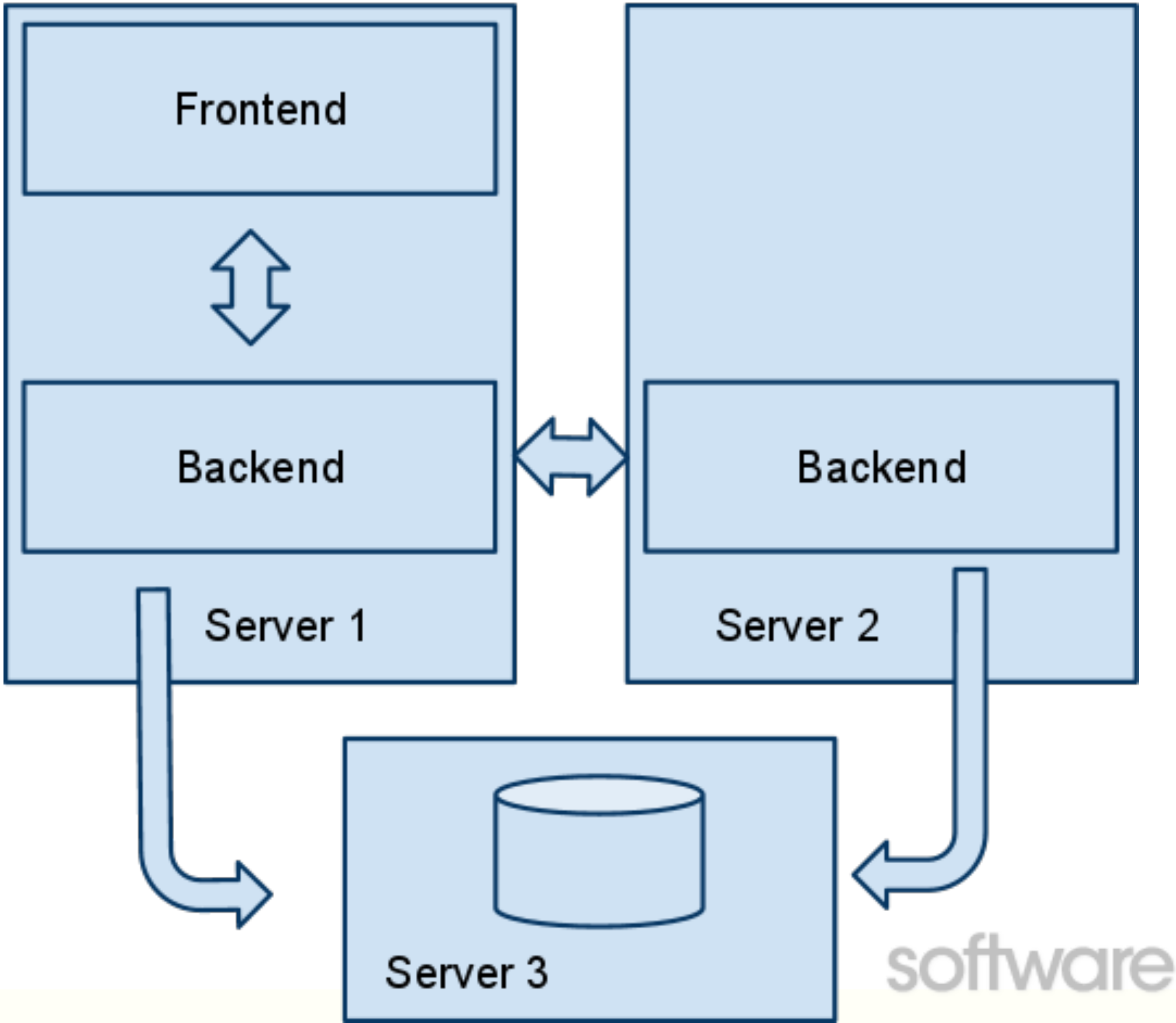
What about the DB?

Data Base

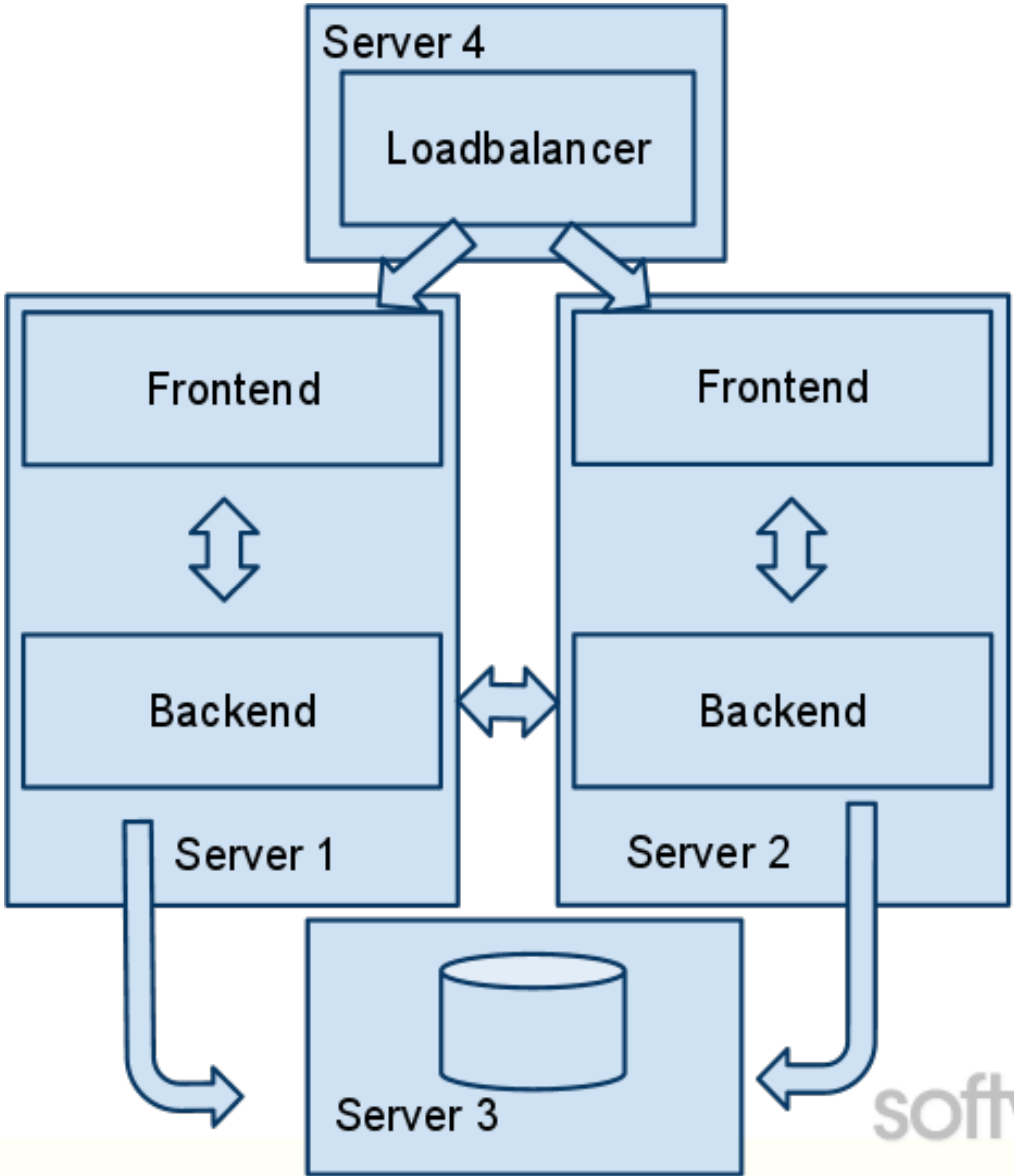


Data Base

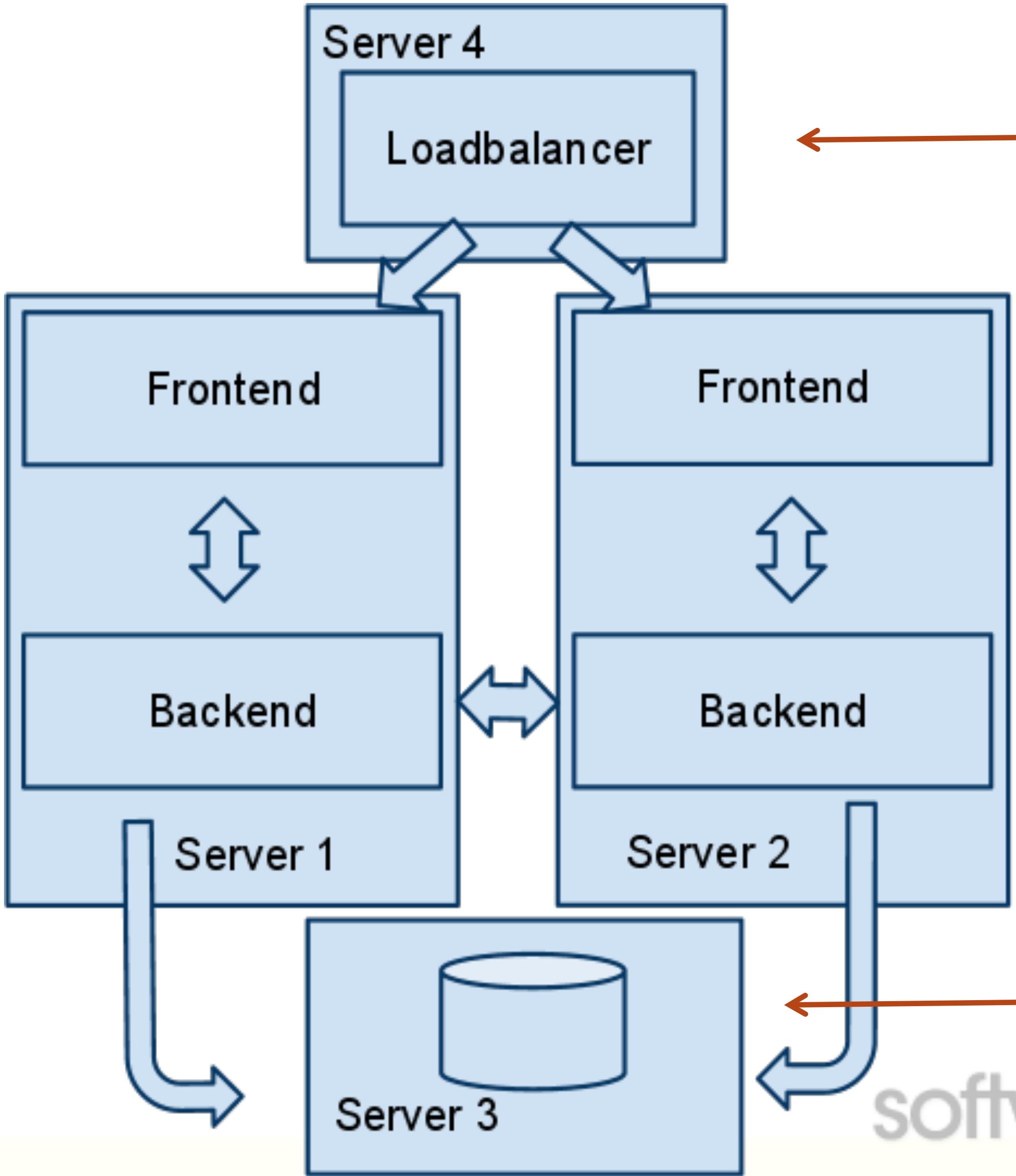
What about scalibility and HA of frontend?



Frontend


Server 4
Loadbalancer
Frontend
Frontend
Backend
Backend
Server 1
Server 2
Server 3

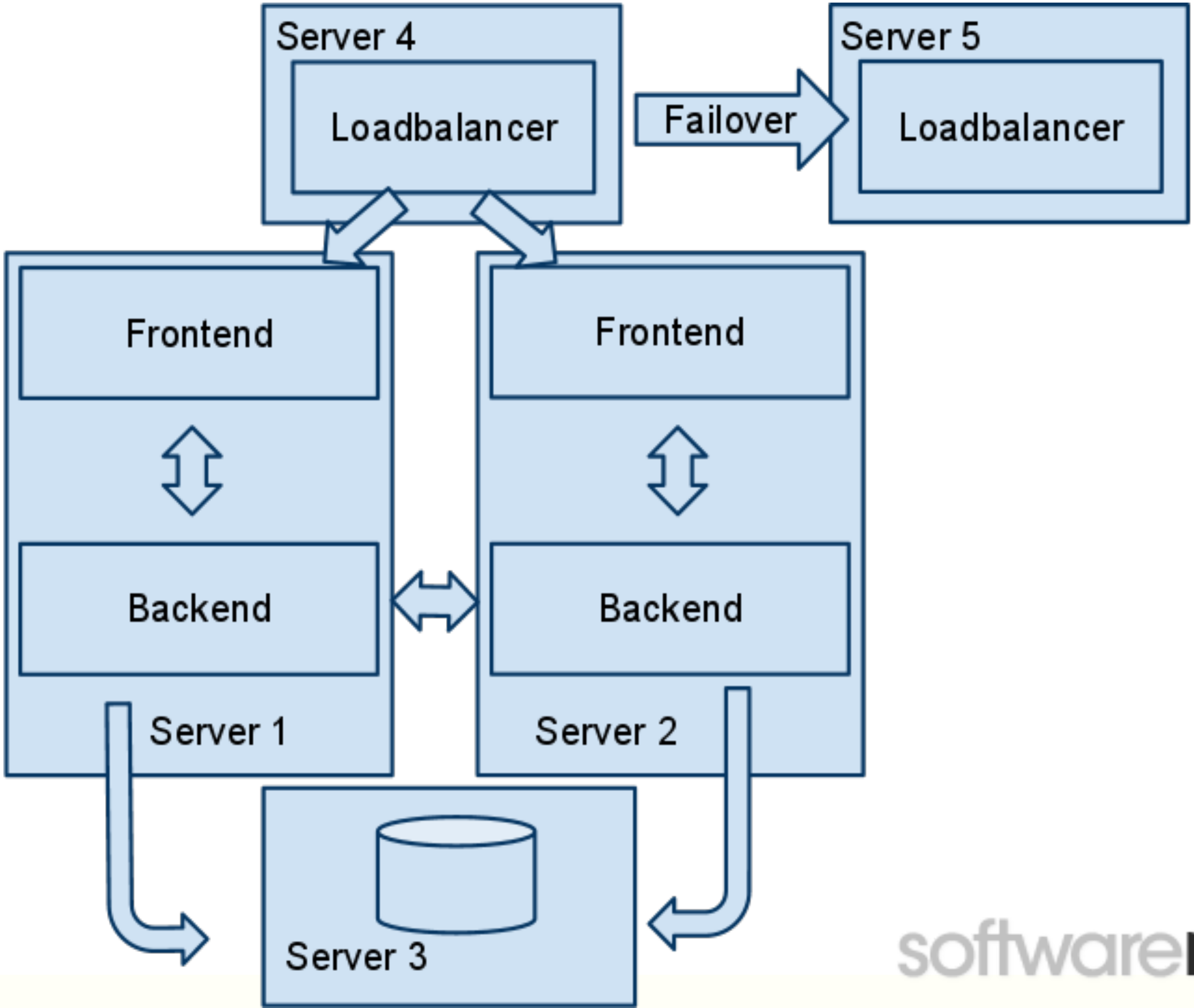
Frontend



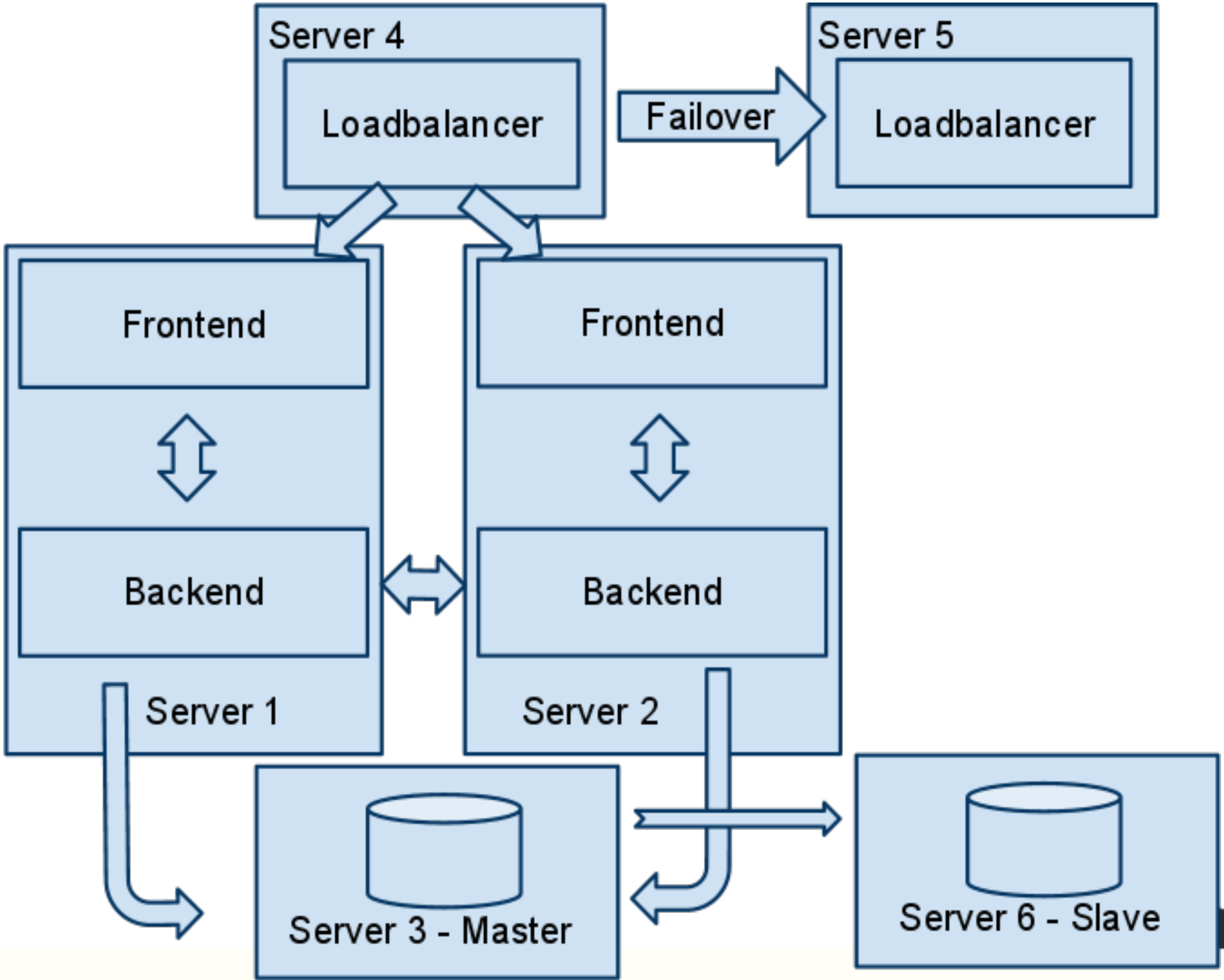
SPOF

SPOF

Loadbalancer



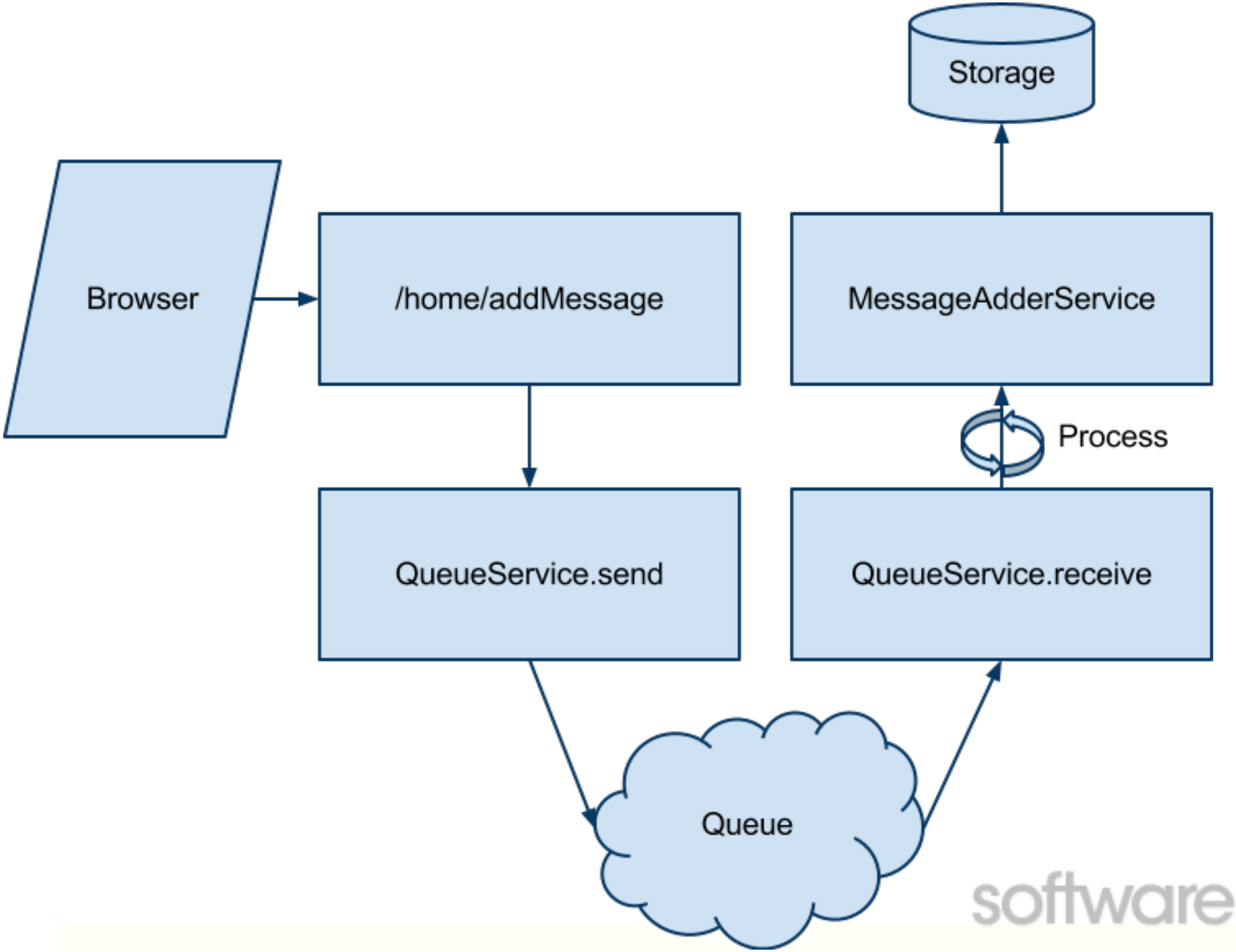
DB


Server 4
Loadbalancer
Failover
Server 5
Loadbalancer
Frontend
Backend
Server 1
Frontend
Backend
Server 2
Server 3 - Master
Server 6 - Slave

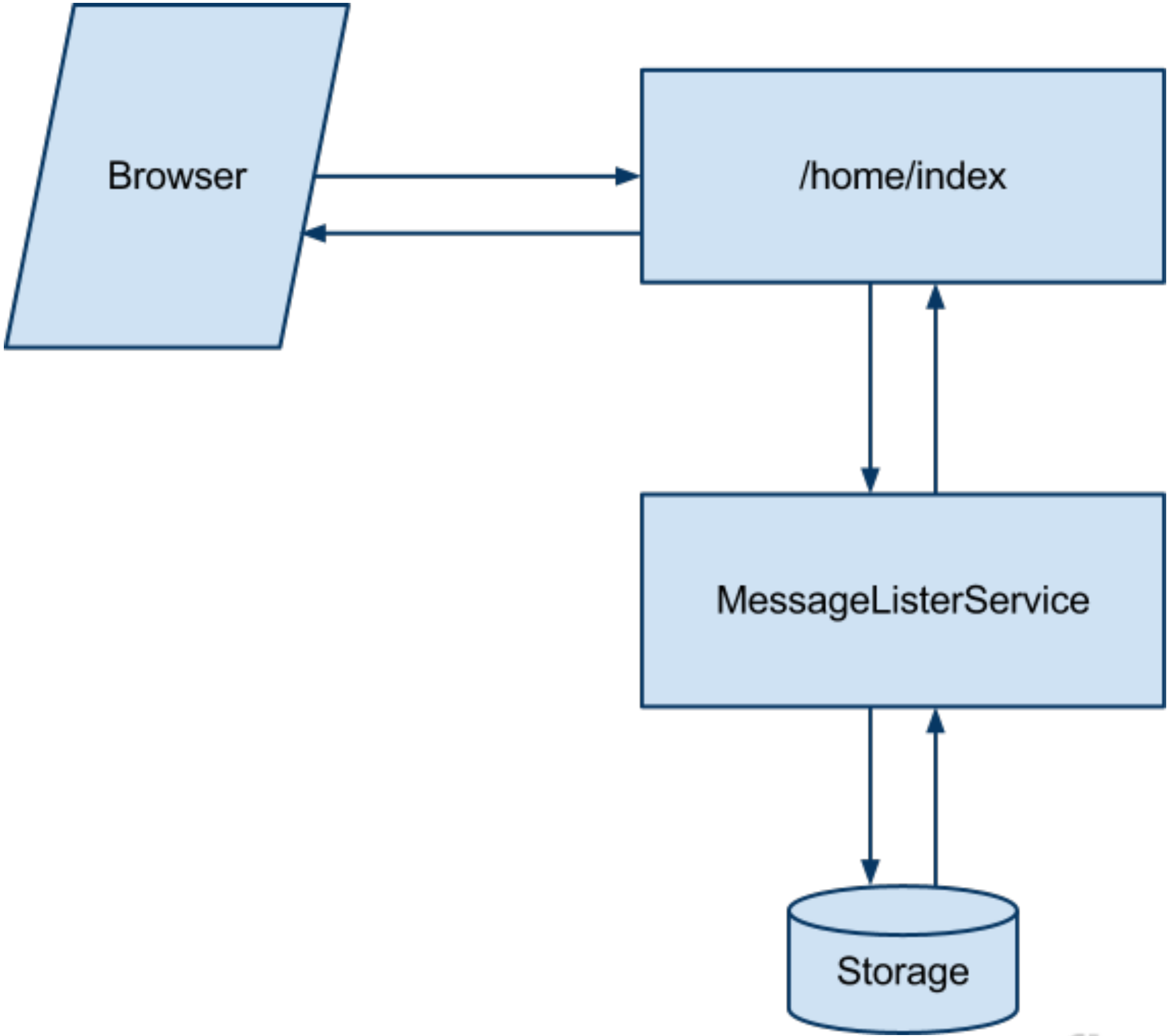
Effect?

- 6 servers
- Lots of spent Rands
- It still isn't very safe (we rely on one failover for each component)

Our application- write



Out application - read



Demo - locally

- JMS
- Hibernate
- H2

Servers – EC2

- New server in couple minutes
- API
- Lots of ready system images
- You can create your own images (AMI)
- Servers in different sizes and flavors
  - ie. small server 0.095c/h ~= 68.4 USD/month
- 6 regions, few availability zones in each
- SLA 99.95%

Servers – ELB

- You can attach as many EC2 servers as you want
- Work per region
  - Which means couple of availability zones
- Integration with Route53 (DNS) – domain support
- Sticky sessions
  - For example using JSESSIONID cookie

Persistence – SimpleDB

- NoSQL
- Key-value storage (multiple values per key)
- Eventual consistency, but:
  - Consistent read
  - Conditional put (~ transactions)
- SQL-like language for queries, but:
  - No JOINs
  - Query on N attributes will perform N queries and combine
- Everything is a String

Queues- SQS

- Pull-type message receiving
- No transactions
- Message will be delivered again, If it’s not deleted after receive
- Messages are stored on many servers, so there is a slight chance it can be delivered more then once

Demo – Amazon

- SQS instead of JMS
- SDB instead of Hibernate
- EC2 instead of Macbook Pro
- All the code available on GitHub!
  - https://github.com/szimano/jozi-aws-demo

Links

- **https://github.com/szimano/jozi-aws-demo**
- **http://aws.amazon.com**
- https://www.jbison.com
- https://www.circulardms.com

Thank you

- Tomasz Szymański
  - tomasz.szymanski@softwaremill.com
  - @szimano

Oh BTW - we are hiring in SA ;-)