

# Comet

what's next after Ajax?

-- and a little about Erlang

# Contents

- comet, why?
- comet, what?
- problem?
- solution!
- model?
- little demo.
- what's next?
- Q&A

Ajax is a Better Web, But:

far away from a Regular-Client.

comet, why?

Because HTTP is **Stateless**, so, it's:

- One-Way communication
- Non-Real-Time

-- Low-Interactive!

comet, why?

What we need is:

- Two-Way
- Real-Time
- and Still Web-Browser Based

--Hi-Interactive!

it's why we need comet.

comet, why?

So we can do these:

- Web Chat
- Web MMO-Game
- Web Live data
- Web Cooperation
- .....

it's a better WEB!

comet, why?



There are 3 ways to implement comet:

- Client Pull
- Hybrid
- Server Push

no one is new technology!

comet, what?

## Client Pull

- Polling
- Hidden IFrame
- Meta refresh tag
- AJAX refresh

the key point is: same web-server, but a loop client.

comet, what?



## Hybrid

- Long Polling
- The Google Way

the key point is: a loop client, a time-limited server.

comet, what?

## Server Push

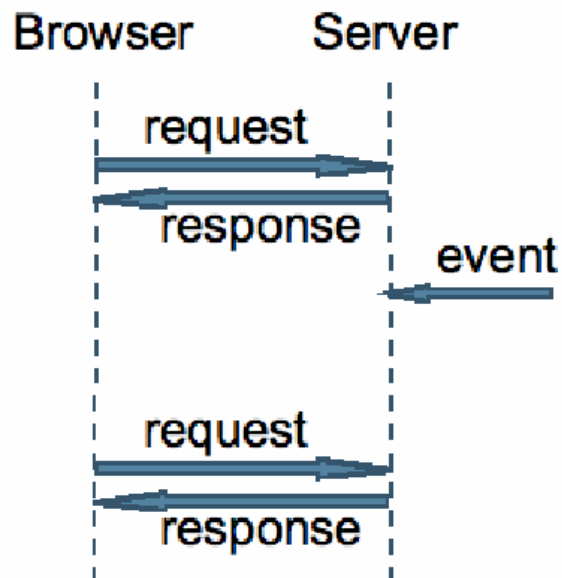
- Steaming
- Keep-Live/Long connection
- Multipart-Replace mime type
- Non-End page

the key point is: a long-live connection.

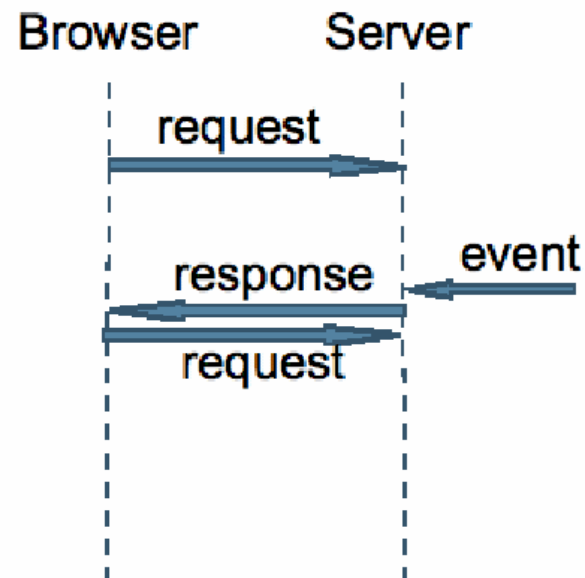
comet, what?

long-term HTTP connections

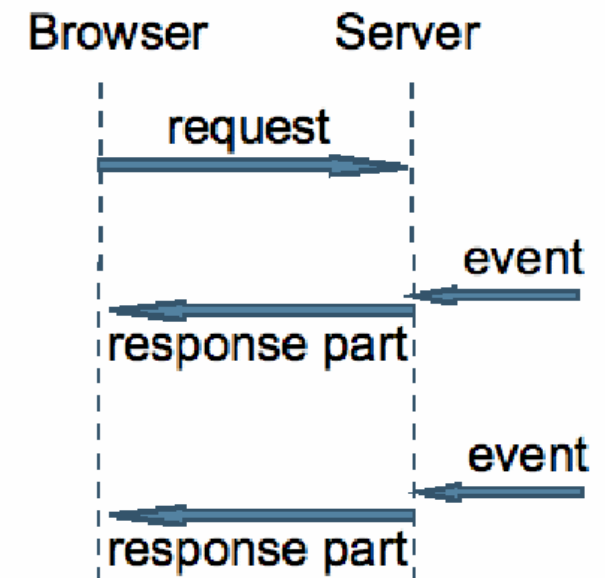
### Ajax (Polling)



### Ajax Push (Long Poll)



### Ajax Push (Streaming)



comet, what?

There are all good methods, but:

they all has a fatal problem.

- The Limits on "Concurrent Connections"

- The deeper problem is "leak of a model"

problem?

LAMP -> ~c2k - c5k

Jetty -> ~c5k?

LightHttpd -> ~c10k

Nginx -> ~c10k

but what we need is c1M!

problem?

And, more important problem is:

-- We Need a new MODEL!

problem?



the solution is: the Erlang way.

- 1M connections,
- 1M/3G RAM, ~8k~40k per connection,
- ~5% CPU

Realistic!

solution

and the model is:

## Event-Driven WEB!

- Web As a UI
- UI -> **event** -> Server
- Server -> **event** -> UI

Just like the traditional UI programming model!

model

# Little Demo

funny time -- a tiny web game

a new type of web application!  
javascript is more important!

- protocol friendly?
- developer friendly?
- programming language friendly?

Go Build Some **Real-Time** Web-  
Apps!

what's next?

# Q&A

Erlounge III @ shanghai

2008.12.20 - 21

[erlang-china.org](http://erlang-china.org)

[jackyz.zhao](mailto:jackyz.zhao)