Large-scale Game Messaging in Erlang at IMVU

Jon Watte

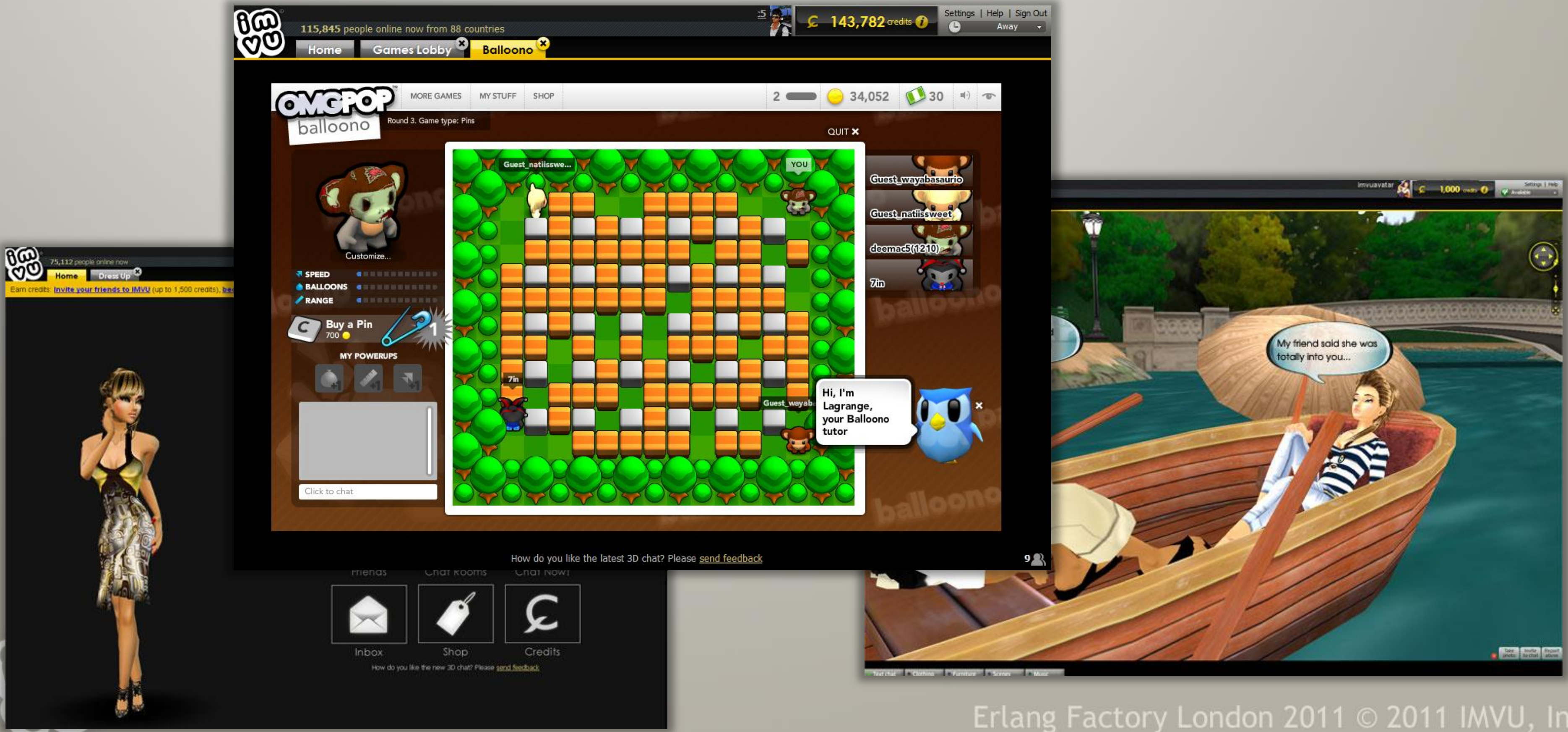
Technical Director, IMVU Inc

@jwatte / #erlangfactory

Presentation Overview

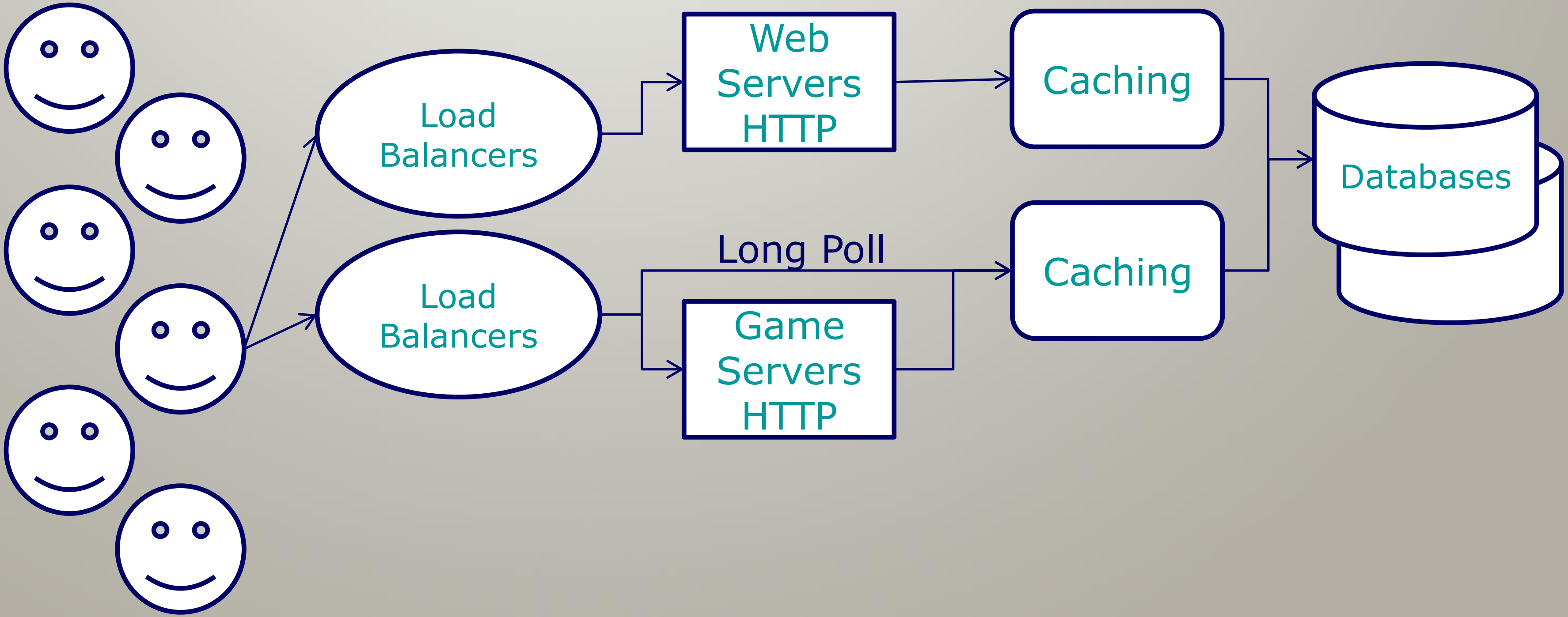
- Describe the problem
  - Low-latency game messaging and state distribution
- Survey available solutions
  - Quick mention of also-rans
- Dive into implementation
  - Erlang!
- Discuss gotchas
- Speculate about the future

From Chat to Games



115,845 people online now from 88 countries
143,782 credits
Settings | Help | Sign Out
Away
Home
Games Lobby
Balloono
OMGPOP balloono
MORE GAMES
MY STUFF
SHOP
34,052
30
Round 3. Game type: Pins
QUIT
Customize...
SPEED
BALLOONS
RANGE
Buy a Pin 700
MY POWERUPS
Click to chat
Guest_natiisswe...
YOU
Guest_wayabasaurio
Guest_natiissweet
deemac5(1210)
7in
Hi, I'm Lagrange, your Balloono tutor
How do you like the latest 3D chat? Please send feedback
75,112 people online now
Dress Up
Earn credits: Invite your friends to IMVU (up to 1,500 credits)
Inbox
Shop
Credits
How do you like the new 3D chat? Please send feedback
1,000 credits
My friend said she was totally into you...

Context


Load Balancers
Load Balancers
Web Servers HTTP
Long Poll
Game Servers HTTP
Caching
Caching
Databases

What Do We Want?

CATERING PROPOSITION MENU
Pizzeria Bellisima
Appetizers

Cold antipasto size for 10-12 ppl	35$
Caesar Sld size 8-10 ppl	35$
Mix Sld size 8-10 ppl	30$
Veggie Platter (carrots,celery, yomatoes,broccoli)	30$
Chicken Stripes 50 pcs	40$
Mozzarella Sticks 50 pcs	35$
Chicken Wings BBQ 50 pcs	40$
Chicken Hot Wings 50 pcs	40$
Fish sticks 50 pcs	35$

Hot Dishes

Backed Ziti size 20 ppl	80$
Penne ala Vodka 20ppl	60$
Spaghetti Alfredo 20 ppl	60$
Cheese or Meat Ravioli size 20 ppl	70$
Cheese or Meat Tortellini size 20ppl	70$
Meat or Veggie Lasagna size 20ppl --	80$
Chicken 30 pcs	90$
Tilapia 30 pcs	110$
Choice of yellow rice 20$ or FF for 20ppl	40$

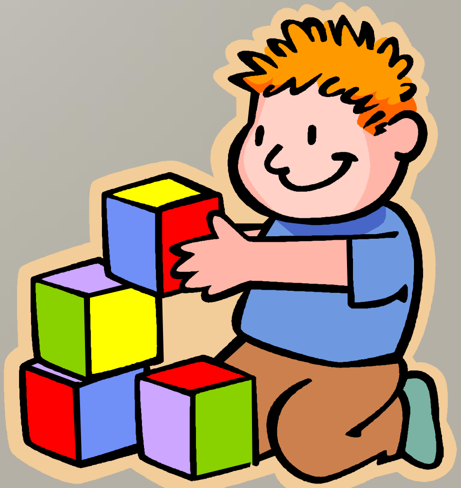
Desserts
Gelato
NY Cheese Cake
Tiramisu
Home Made Cookies
Chocolate Cake

Beverages
Coffee and Tea together
Soda choice (coke, sprite, pepsi, Sunkist) 2L
Juices Tropicana (Orange, Mix)

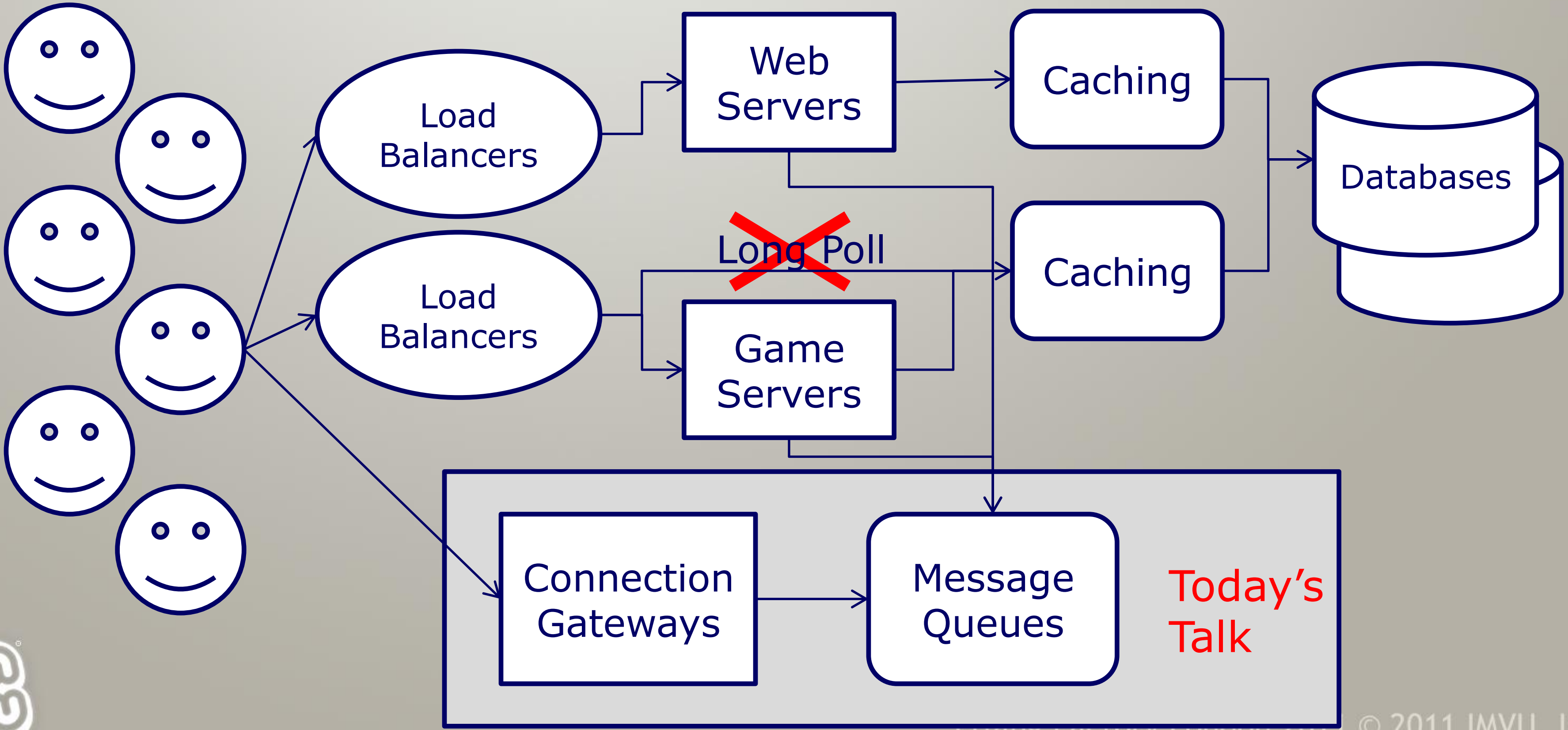
- **Any-to-any** messaging with ad-hoc structure
  - Chat; Events; Input/Control
- **Lightweight** (in-RAM) state maintenance
  - Scores; Dice; Equipment

New Building Blocks

- **Queues** provide a sane view of distributed state for developers building games
- Two kinds of messaging:
  - **Events** (edge triggered, “messages”)
  - **State** (level triggered, “updates”)
  - Expressed as “mounts”
- **Integrated** into a bigger system



From Long-poll to Real-time



Performance Requirements

- Simultaneous user count:
  - 80,000 when we started
  - 150,000 today
  - 1,000,000 design goal
- Real-time performance (the main driving requirement)
  - Lower than 100ms end-to-end through the system
- Queue creates and join/leaves (kill a lot of contenders)
  - >500,000 creates/day when started
  - >20,000,000 creates/day design goal

Also-rans: Existing Wheels

- AMQP, JMS: Qpid, Rabbit, ZeroMQ, BEA, IBM etc
  - Poor user and authentication model
  - Expensive queues
- IRC
  - Spanning Tree; Netsplits; no state
- XMPP / Jabber
  - Protocol doesn't scale in federation
- Gtalk, AIM, MSN Msgr, Yahoo Msgr
  - If only we could buy one of these!

Our Wheel is Rounder!

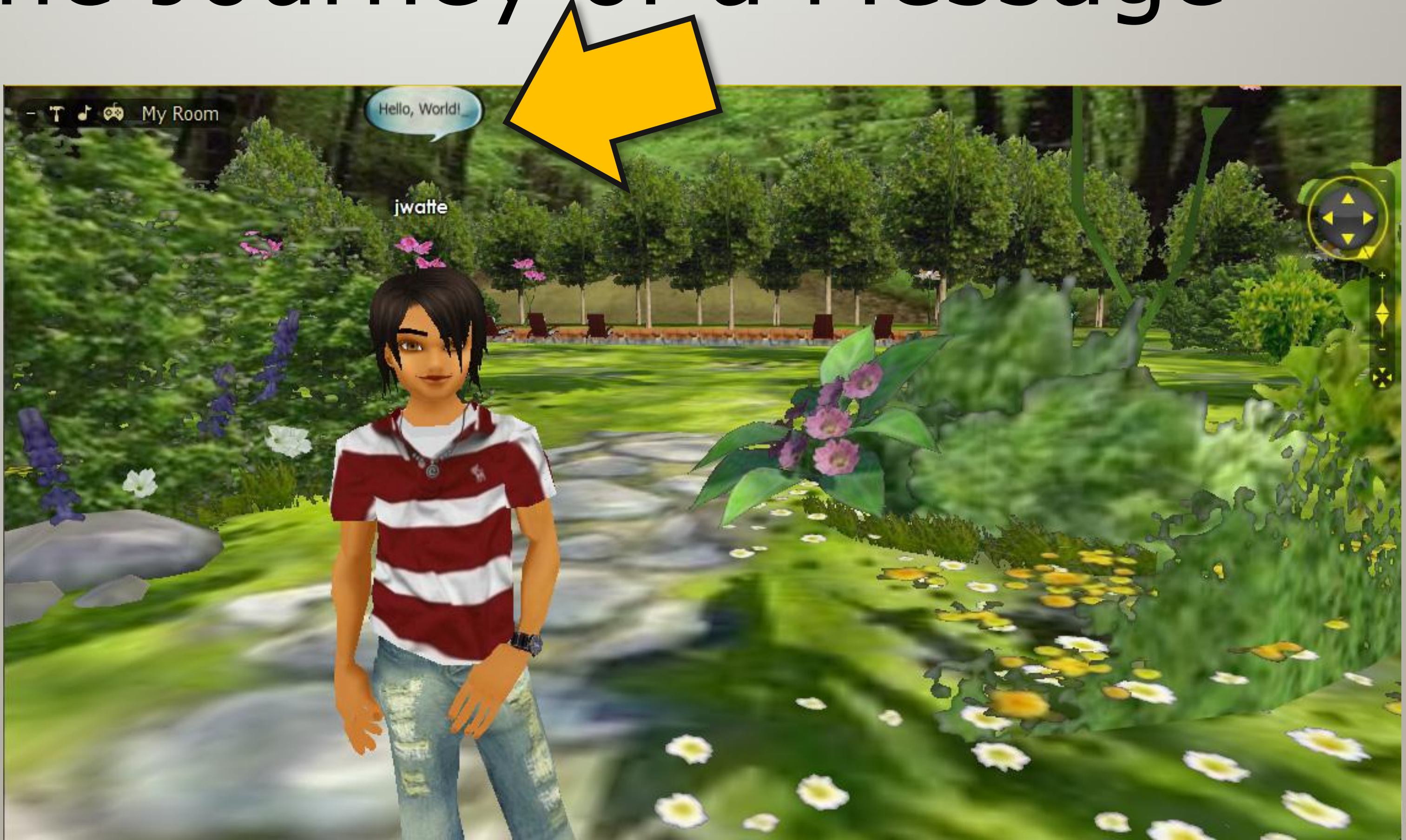
- Inspired by the 1,000,000-user mochiweb app
  - http://www.metabrew.com/article/a-million-user-comet-application-with-mochiweb-part-1
- A purpose-built general system
- Written in Erlang

- @jwatte / #erlangfactory

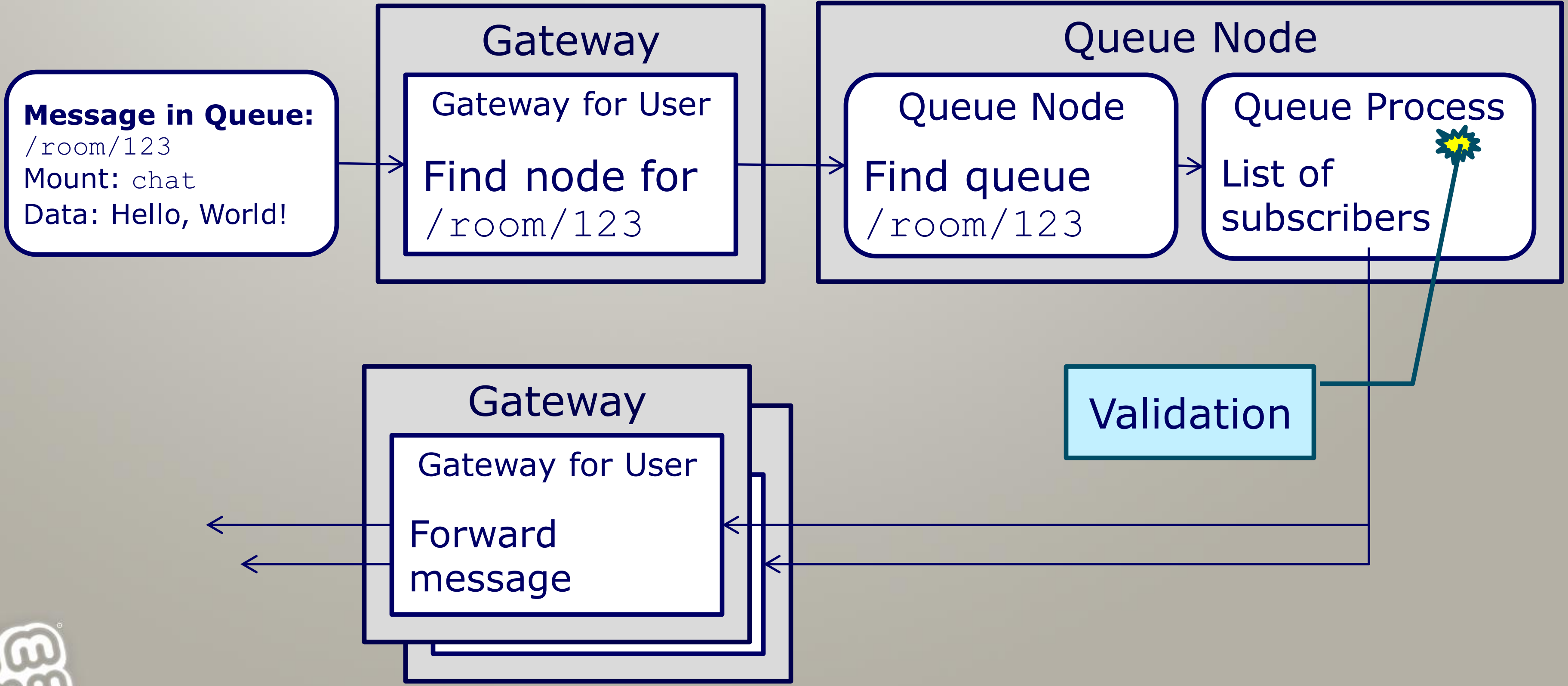
Section: Implementation

- Journey of a message
- Anatomy of a queue
- Scaling across machines
- Erlang

The Journey of a Message



The Journey of a Message



Anatomy of a Queue

Queue Name: `/room/123`

Mount
Type: message
Name: `chat`

User A: I win.
User B: OMG Pwnies!
User A: Take that!
…

Mount
Type: state
Name: `scores`

User A: 3220
User B: 1200

Subscriber List

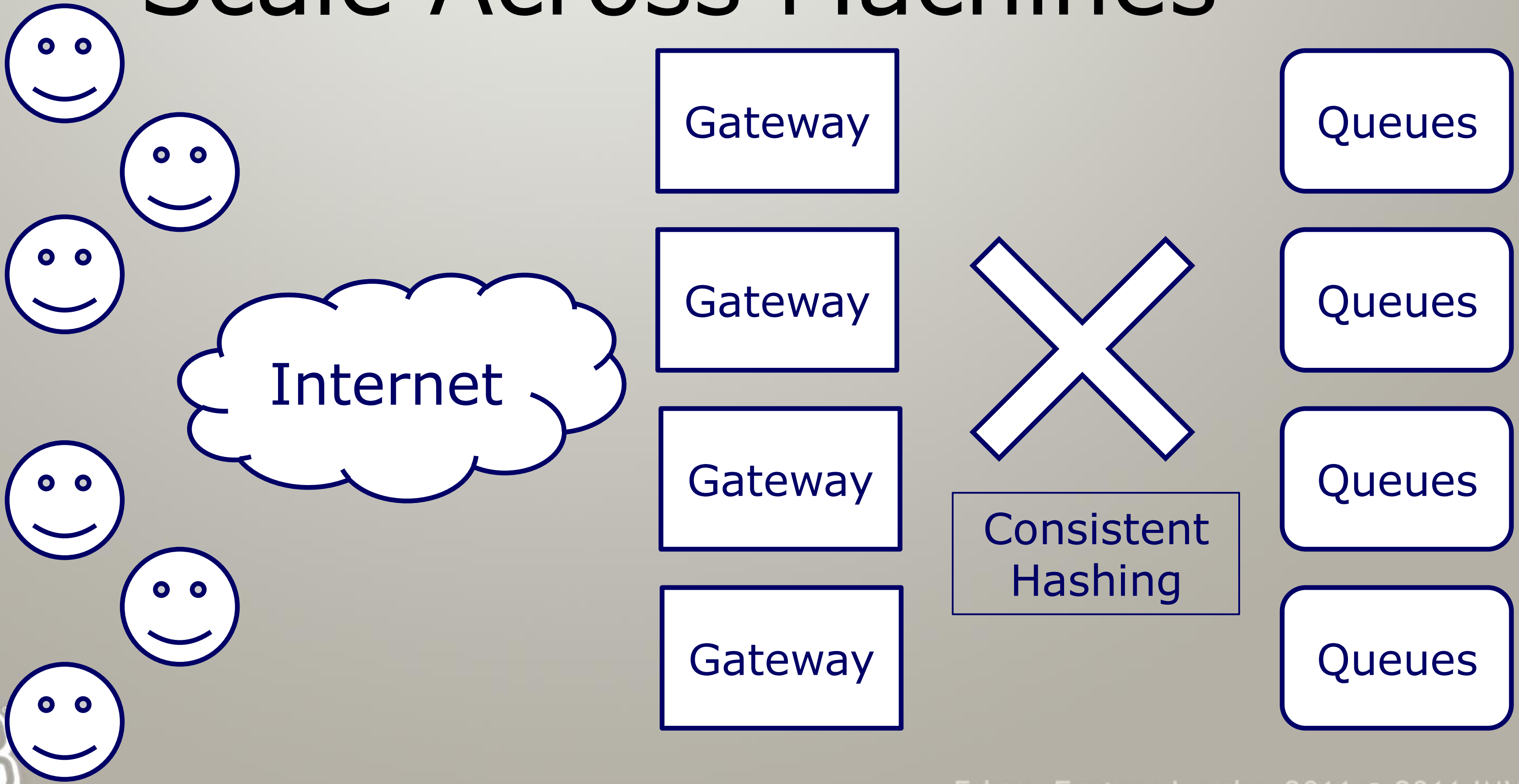
User A @ Gateway C

User B @ Gateway B

A Single Machine Isn’t Enough

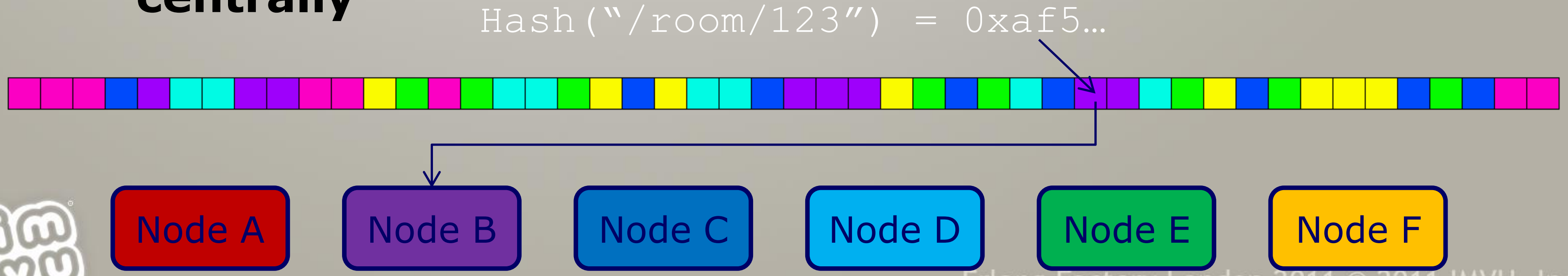
- 1,000,000 users, 1 machine?
  - 25 GB/s memory bus
  - 40 GB memory (40 kB/user)
  - Touched twice per message
  - **one message** per is **3,400 ms**
- @jwatte / #erlangfactory

Scale Across Machines



Consistent Hashing

- The Gateway maps queue name -> node
- This is done using a **fixed hash function**
- A prefix of the output bits of the hash function is used as a look-up into a table, with a minimum of **8 buckets per node**
- Load differential is 8:9 or better (down to 15:16)
- Updating the map of buckets -> nodes is **managed centrally**



Consistent Hash Table Update

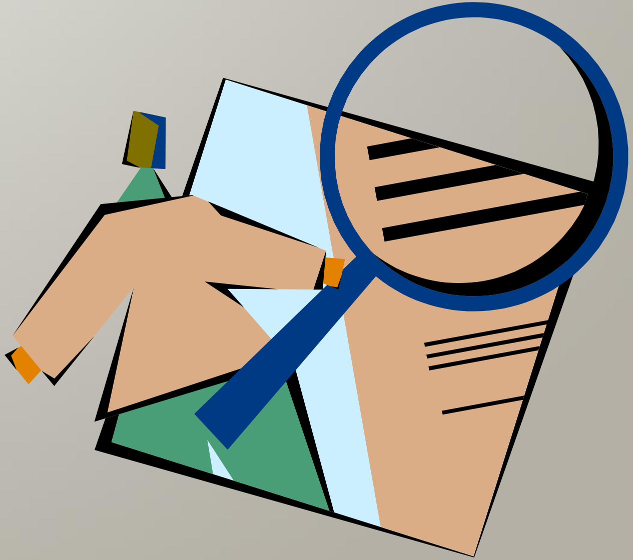
- **Minimizes** amount of data shifted
- If nodes have more than 8 buckets, **steal 1/N** of all buckets from those with the most and assign to new target
- If not, **split each bucket**, then steal 1/N of all buckets and assign to new target

Erlang

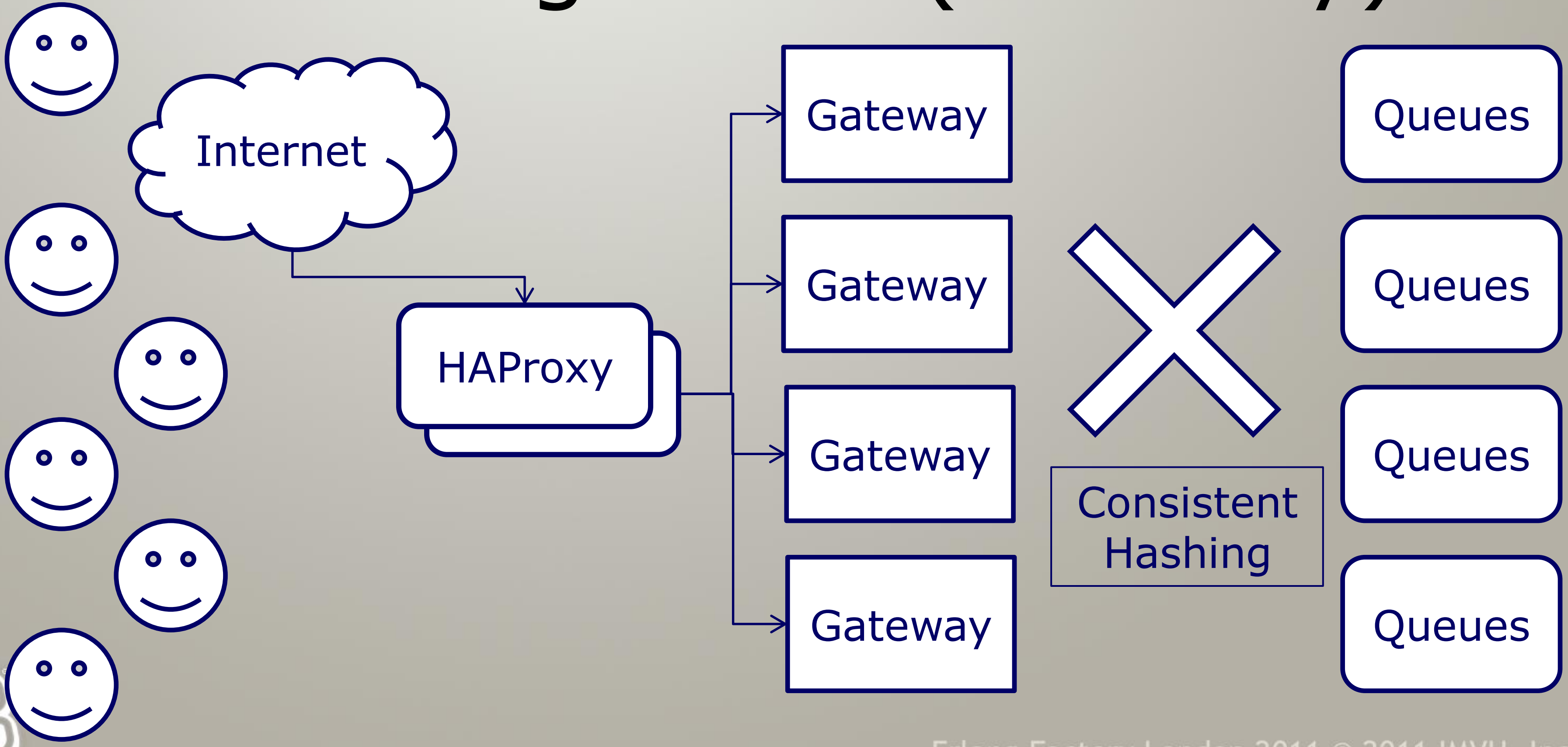
- Develope switches
  - Re
- Pro
  - 25
- Paralle
  - Erlang reads
- (Almost)
  - No data ra
  - Each proces age collected

Section: Details

- Load Management
- Marshalling
- RPC / Call-outs
- Hot Adds and Fail-over
  - The Boss!
- Monitoring



Load Management (HAProxy)



Marshalling (protobuf)

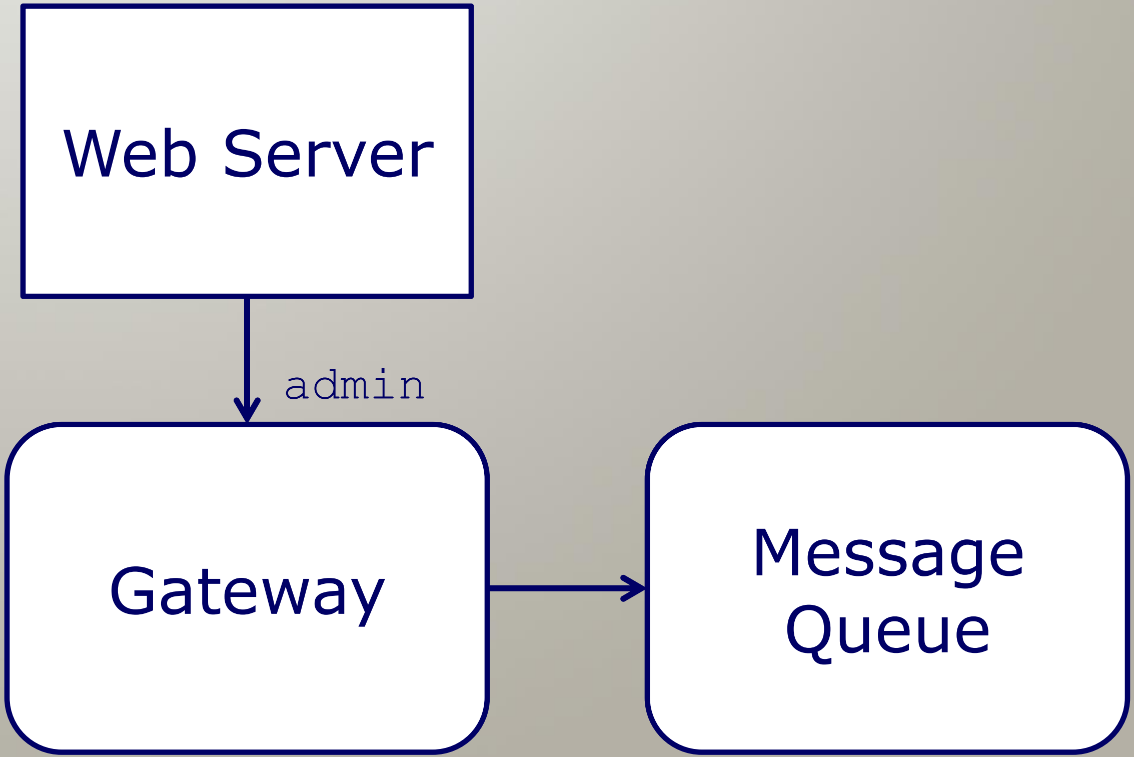
```
message MsgG2cResult {
    required uint32 op_id = 1;
    required uint32 status = 2;
    optional string error_message = 3;
}
```

RPC (HTTP + JSON)

PHP

HTTP + JSON

Erlang

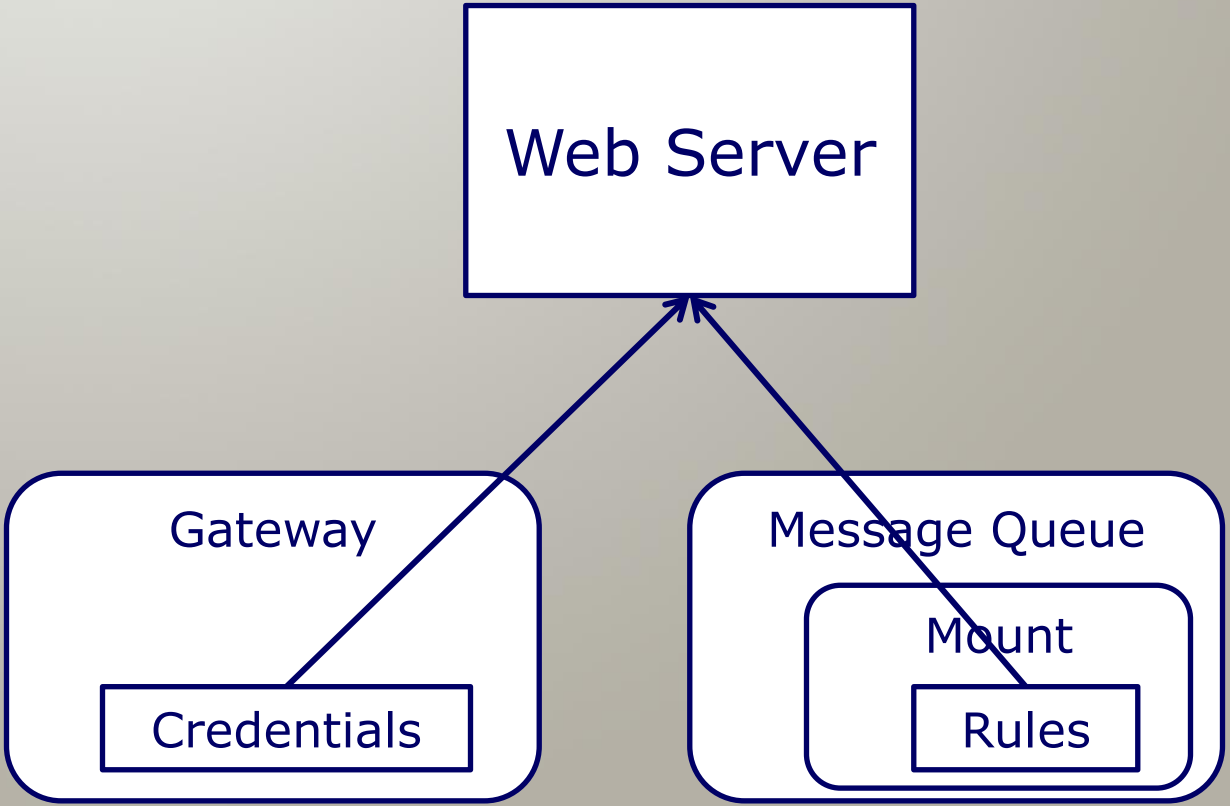


Call-outs (HTTP + JSON)

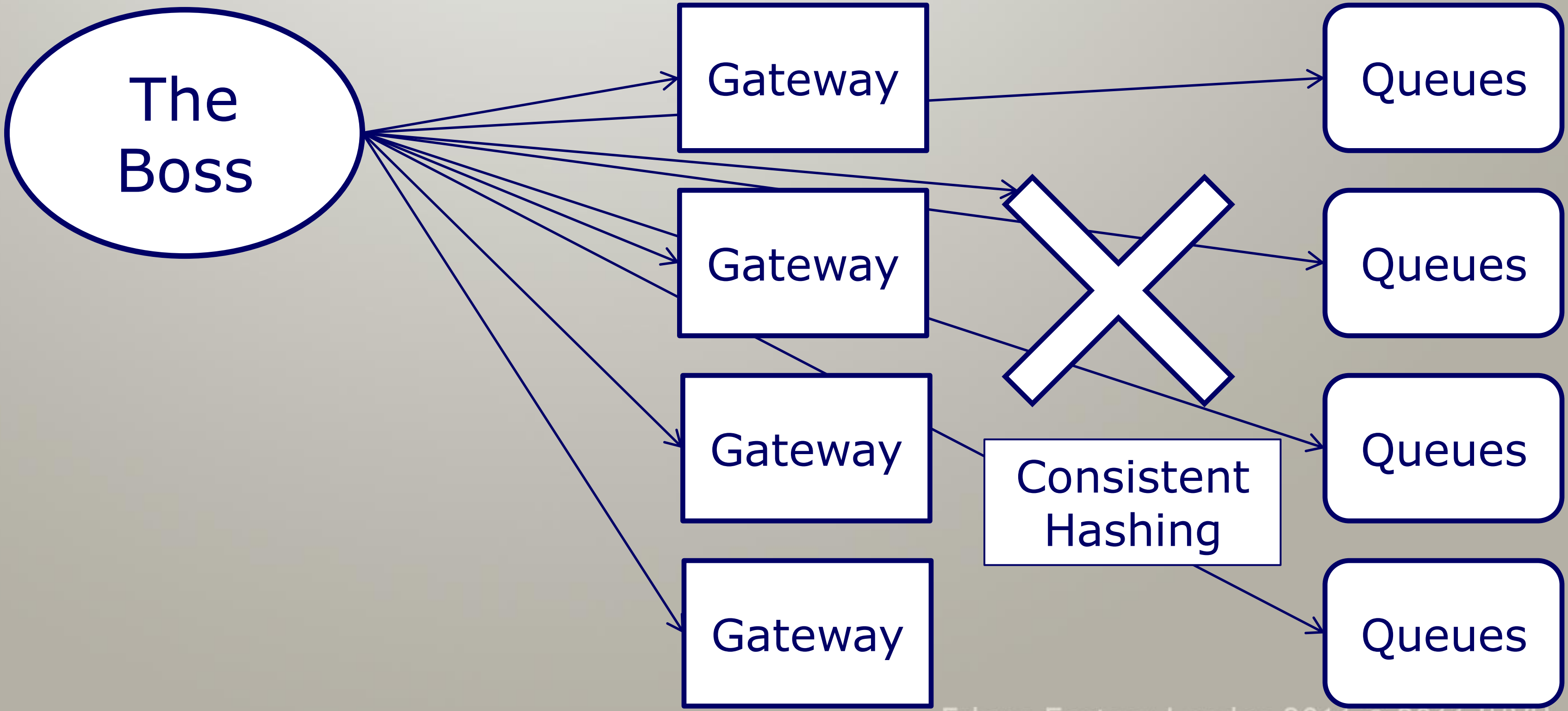
PHP

HTTP + JSON

Erlang

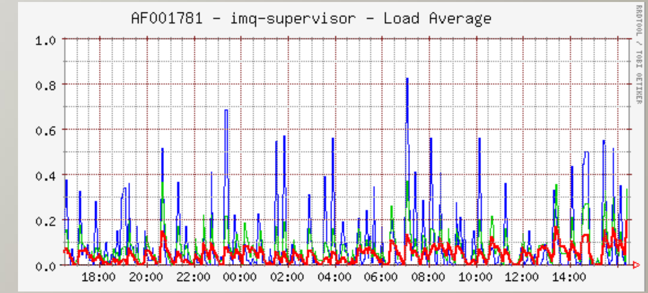


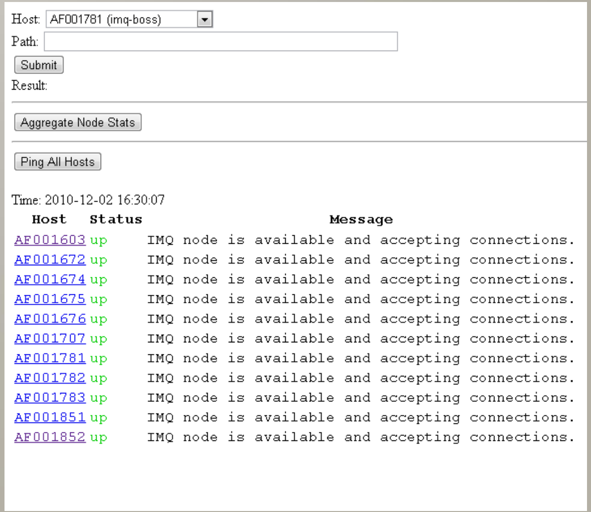
Management


The Boss
Gateway
Gateway
Gateway
Gateway
Queues
Queues
Queues
Queues
Consistent Hashing

Monitoring

- Example counters:
  - Number of connected users
  - Number of queues
  - Messages routed per second
  - Round trip time for routed messages
    - Distributed clock work-around!
  - Disconnects and other error events




Host: AF001781 (imq-boss)
Path:
Submit
Result:

Aggregate Node Stats

Ping All Hosts

Time: 2010-12-02 16:30:07

```
 Host     Status                Message
AF001603 up      IMQ node is available and accepting connections.
AF001672 up      IMQ node is available and accepting connections.
AF001674 up      IMQ node is available and accepting connections.
AF001675 up      IMQ node is available and accepting connections.
AF001676 up      IMQ node is available and accepting connections.
AF001707 up      IMQ node is available and accepting connections.
AF001781 up      IMQ node is available and accepting connections.
AF001782 up      IMQ node is available and accepting connections.
AF001783 up      IMQ node is available and accepting connections.
AF001851 up      IMQ node is available and accepting connections.
AF001852 up      IMQ node is available and accepting connections.
```

Section: Problem Cases

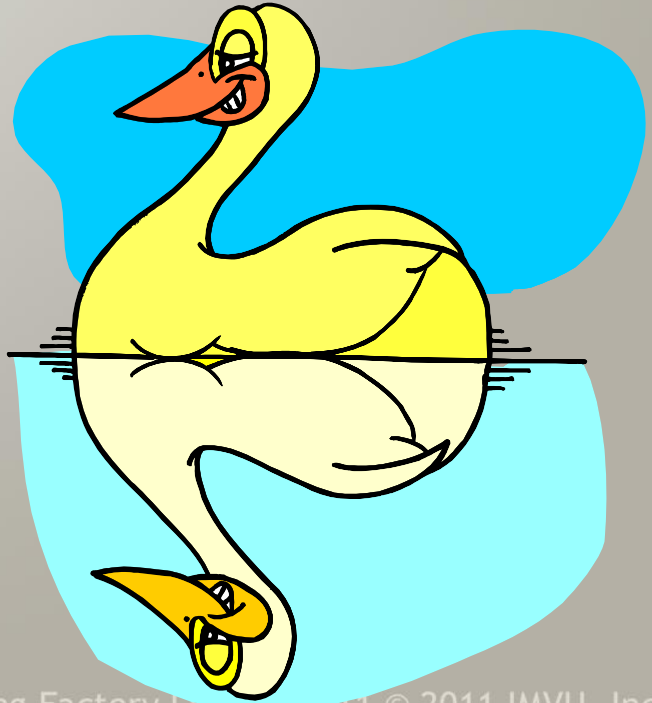
- User goes silent
- Second user connection
- Node crashes
- Gateway crashes
- Reliable messages
- Firewalls
- Build and test

User Goes Silent

- Some TCP connections will stop (**bad WiFi**, firewalls, etc)
- We use a **ping message**
- Both ends separately detect ping failure
  - This means one end detects it before the other

Second User Connection

- **Currently connected user** makes a **new connection**
- To **another gateway** because of load balancing
- A **user-specific queue** arbitrates
- Queues are serialized: there is always a winner



Node Crashes

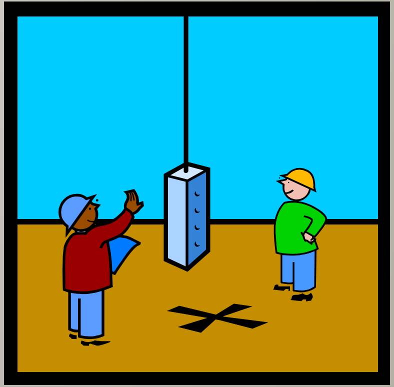
- **State is ephemeral**
  it's lost when machine is lost
- A user "**management queue**"
  contains all subscription state
- If the home queue node dies,
  the user is logged out
- If a queue the user is subscribed to dies, the
  user is auto-unsubscribed (client has to deal)

Gateway Crashes

- When a gateway crashes **client will reconnect**
- **History** allow us to **avoid re-sending** for quick reconnects
- The **application** above the queue API **doesn't notice**
- Erlang message send does not report error
  - Monitor nodes to remove stale listeners

Build and Test

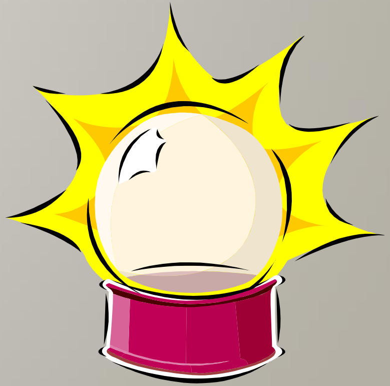
- Continuous Integration and Continuous Deployment
  - Had to build our own systems
- Erlang In-place Code Upgrades
  - Too heavy, designed for "6 month" upgrade cycles
  - Use fail-over instead (similar to Apache graceful)
- Load testing at scale
  - "Dark launch" to existing users
- @jwatte / #erlangfactory



Build and Test contd.

- GNU make
  - Auto-discovers everything as */src/*.erl
  - No recursion or autotools
  - Deals with proto -> .erl/.hrl, etc
- Eunit – built-in, easy to write tests
- Erlymock – mocks more complex functions
- Python-based integration test runner
  - Start X queue nodes, Y gateway nodes, …

Section: Future



- Replication
  - Similar to fail-over
- Limits of Scalability (?)
  - M x N (Gateways x Queues) stops at some point
- Open Source
  - We would like to open-source what we can
  - Protobuf for PHP and Erlang?
  - IMQ core? (not surrounding application server)

Q&A

- Questions?
- Survey
  - If you found this helpful, please use a green card
  - If this sucked, don’t use a green card

- @jwatte
- jwatte@imvu.com
- IMVU is a great place to work, and we’re hiring!