

High availability at Heroku

Mark McGranaghham 

about successes and failures

challenge : evolving socio-technical systems

main pbs

- failed deploys
- bad visibility
- cascading feedbacks

incremental deploys

- initial step : on 1.5% nodes
- few weeks to deploy on every node
- dev/beta/all nodes

shipping tool

realtime visibility

- dashboards monitoring
- very critical to see changes in the platform
- seeing weird traffic

service level assertions

- assert(p99_latency < 50ms)
- monitoring the assertions over time
- monitoring right things are happening

flow control & backpressure

- domino effect
- code level flow control

incremental rollout

- delegating with async publish or dropping messages

principle like erlang 

process supervision

no stopping the world

message passing

- productor
- consumers
- brokers

pb : message versionning

write call graph de-synchronization

- write little tickets
- send data later

graph node failures

brokered queuing

- SPOF with brokers
- publish one/subscribe many