



## Bad Machinery: Managing Interrupts Under Load

Dave O'Connor (daveoc@), Storage SRE

# Whut

- Google SRE since 2004
  - GMail
  - Google Apps Console
  - Reader
  - Google Accounts
  - Google Analytics
  - BigTable
  - Colossus
  - Spanner
  - Logs
  - MySQL

# Interrupts vs. Projects

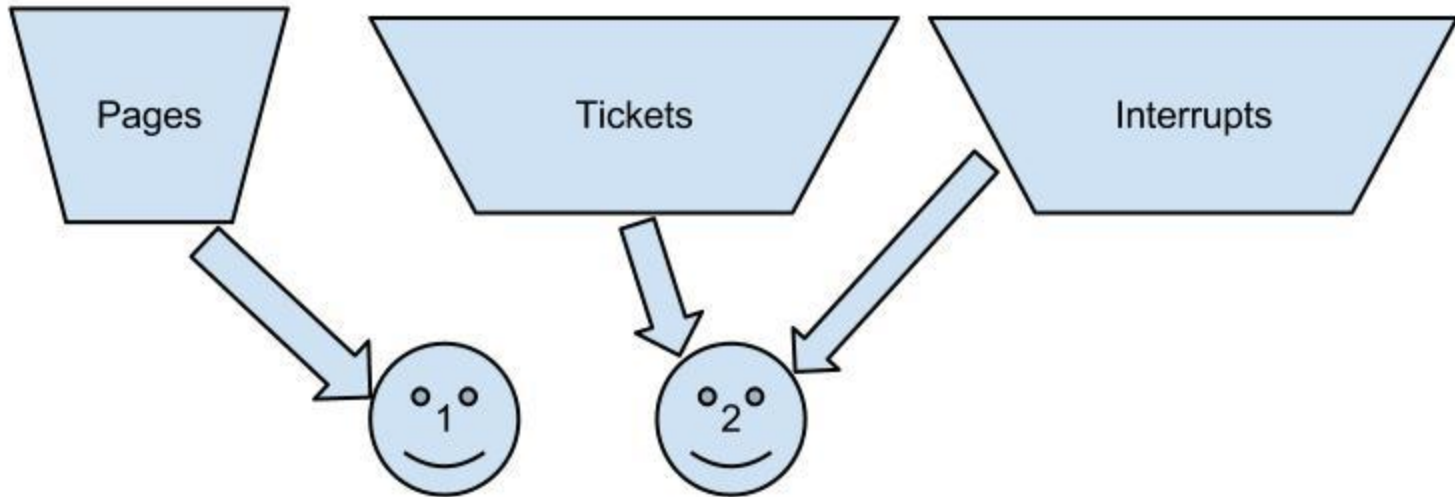
## Projects

- Contribute to development of service medium/long term.
- Highly Creative.
- (Theoretically) fun to do.

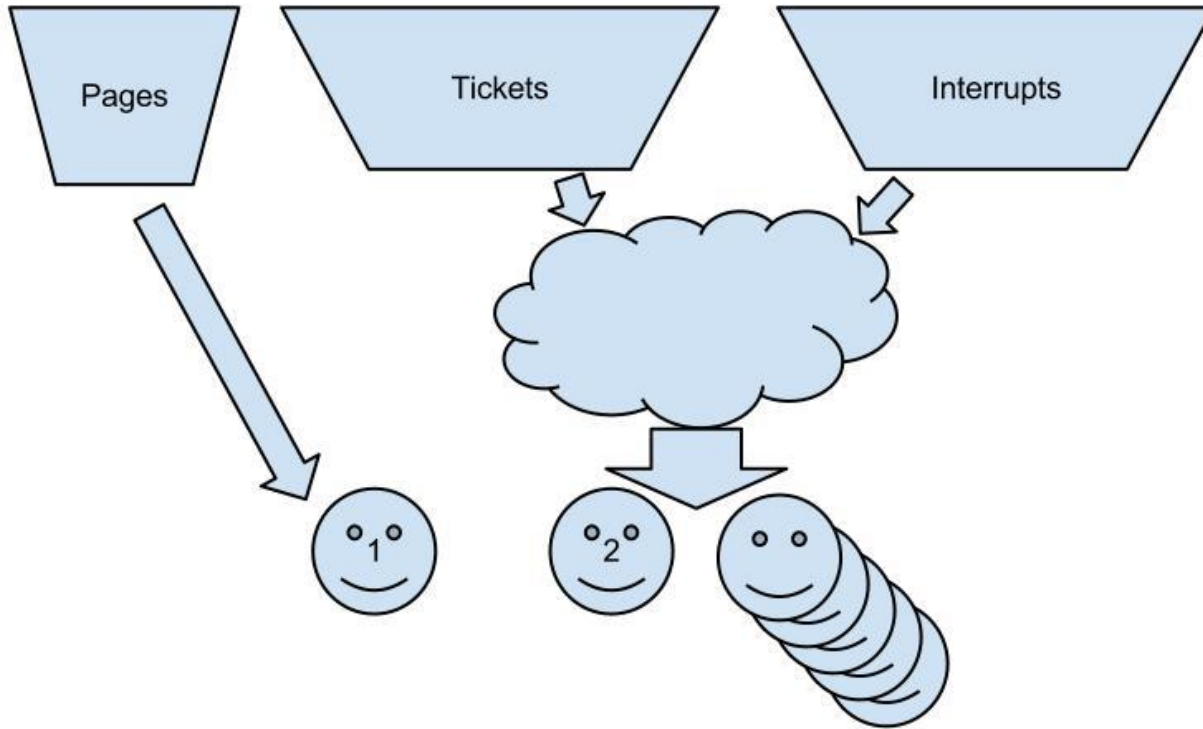
## Interrupts

- More tactical, immediate fixes.
- Usually not so much.
- More often not.

# How Interrupts work in Theory



# How Interrupts work in Practice



## Why the model breaks down

- Interrupt load is too much for 1+2.
- Interrupts are specialised to a person or subset of the team.
- Intentionally.

# Antipatterns

- "The Gauntlet"
- "The Busy Worker"
- "The Amazing Disappearing Category"

## Things We Must Discuss

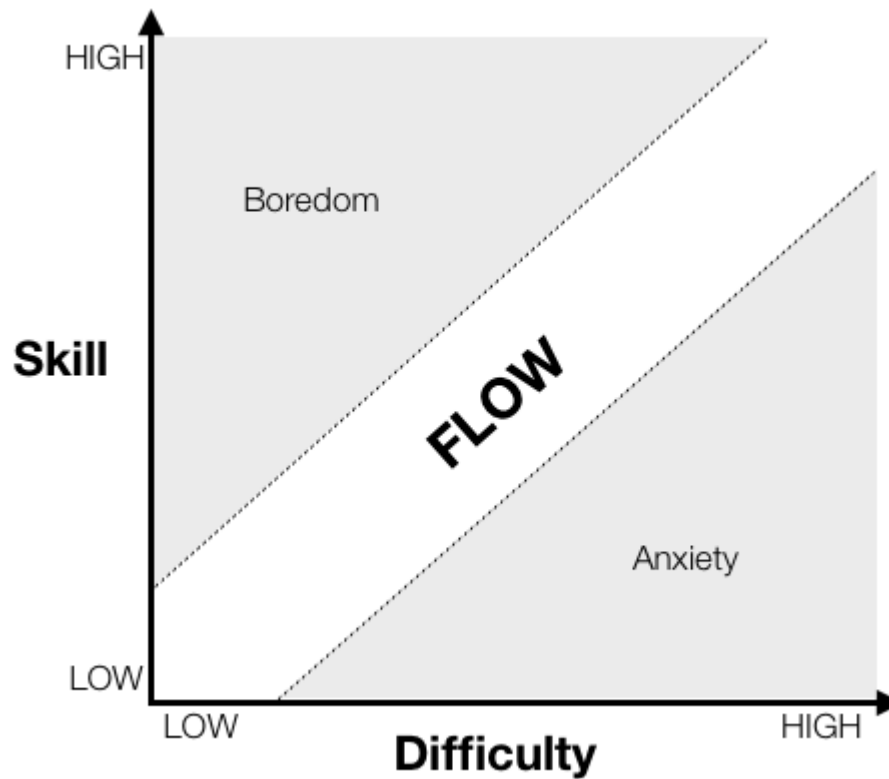
- Context Switches are Hard





## Things We Must Discuss

- Cognitive Flow State is Hard



# Things We Must Discuss

Oncall is a Project

Oncall does not care whether you consider it a priority. It just is.

# Things We Must Discuss

Fairness is Easy to Program

...if you assume people are machines

# Fundamentals

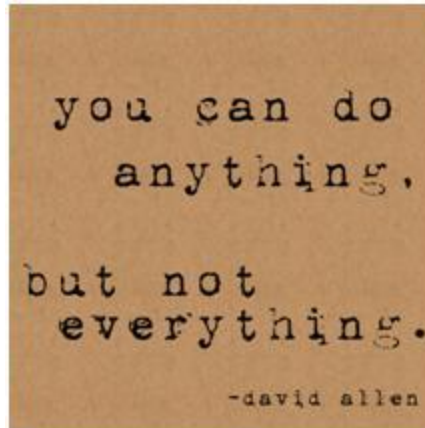
- Humans are Bad Machinery.

*"Humans **are** bad machinery. They get bored, they have processors (and sometimes UIs) that aren't very well-understood, and aren't very efficient"*

- Cognitive Flow is Precious.
- Time should be Polarised. Do one thing well.

## Practicals

- Polarise Time



# Practicals

- Think about Interruptibility



# Practicals

Do For Tickets What You Do For Pages

# Practicals

Respect Yourself



## Other Lessons Learned

- Email alerts are from the past.
- Consensus is nice.
- Policy is as powerful a tool as code.
- The A stands for agreement.

## How Can I Apply this?

- Minimise time the individual can be interrupted.
- Do for Interrupts what you do for Oncall
- Respect the Customer and Respect Yourself.