

*{Muzak playing}*

...

*Supercalifragilisticexpialidocious*

or

*Embracing the Change Freeze*

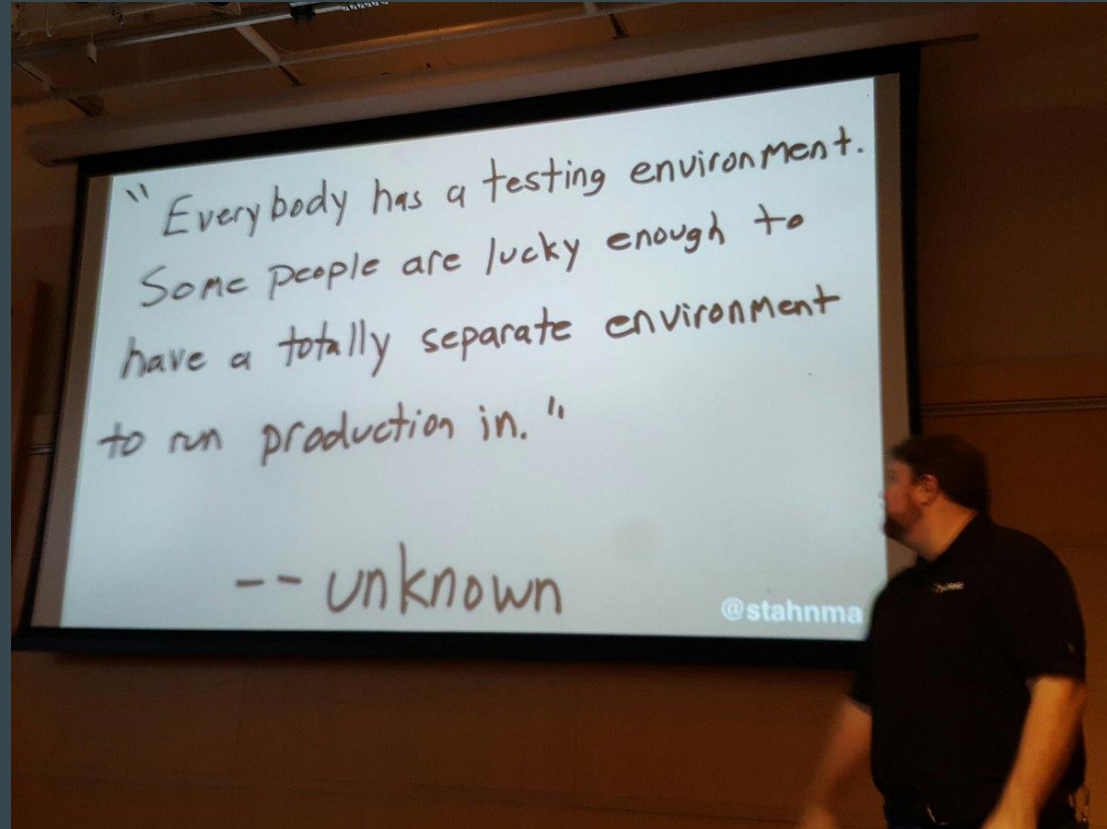
•••

What to ~~say~~ do when you don't know what to ~~say~~ do

by Warren Myers

# Background - Change Management

- What is it?
- Why do I care?
- Real-world examples



# The Types of Work



(hat tip, *The Phoenix Project*)

- Business Projects - Keep Making Money
  - Work that enables the business to do its thing
- Internal Projects - Organization Maintenance, “Keeping the Lights On”
  - Infrastructure improvements
  - Stuff that “just happens”
- Operational Change - Business Streamlining
  - New vendor relationships which bring new tools
  - Process change and improvement
- Unplanned Work - “Firefighting” and Other “Emergencies”
  - Emergencies
    - Real
    - Imagined
  - Unplanned bug fixes
  - Etc

# What's Wrong with Fighting Fires?

- It eats time
- It prevents improvement
- It [usually] doesn't solve the problem *long-term*
  - Amazon has a bad interview question related to this
- It's not [easily] explainable
  - Like finding JVM memory issues
- It's soul-sucking
- **Except** for “real” emergencies
  - Truly unforeseen/unforeseeable events
  - But it still sucks
- Debugging is not fire-fighting



# What is a Change Freeze?

- A time you can't make changes!
  - OK, that wasn't especially helpful
  - But it's true
  - Sorry
- A time you can't make changes *because*
  - Something "important" is happening (or should happen)
  - Regulatory requirements
  - It's time to ship
  - No one is around to fix what breaks
  - It's scheduled



# But, Aren't I "Frozen"?

- Well, kinda
  - Features, "unnecessary" bug fixes, "changes" ... ***In Production***
- But not really
  - Because you can always dev on your local code branch
  - There's always something to learn
  - You aren't nearly as clever as you think you are
    - And cleverness (for the sake of cleverness) is bad - pull Kernighan's Lever
    - But you need to get cleverer - `The "cleverness" required to write and understand intricate code is an acquired mental skill.`
- Unless you are ... in which case
  - Go home
  - Take a hike
  - Go to the pub
  - Keep your phone off

# So What Can I Do?

- Hunt-down outstanding bugs
- Talk to your organization's support team
  - Learn how/when the environment breaks (because it does)
- Read the codebase of other groups
- Socialize outside your team
- Learn the framework you're using
  - I know, you know it already
  - Learn it well enough to *teach* it
  - Be boring
- Be the change you want to see
- Optimize your work
  - But don't over-optimize





# Complaints, Compliments, Concerns ...



# Further Resources

- <http://byrbyr.ga/2> - IT Operations Only Does 4 Things
- <http://byrbyr.ga/3> - The Phoenix Project
- <http://byrbyr.ga/4> - Supercalifragilisticexpialidocious
- <http://byrbyr.ga/5> - ITIL on the pedia of wiki
- <http://byrbyr.ga/6> - More history of the “freeze/unfreeze” model
- <http://byrbyr.ga/7> - Time Management for System Administrators
- <http://byrbyr.ga/8> - Rapid feedback development cycle (spiral model)
- <http://byrbyr.ga/9> - You are responsible for your own learning
- <http://byrbyr.ga/a> - You are your own project manager
- <http://byrbyr.ga/b> - The 10/20/30 rule (which I broke - because this isn't PowerPoint)
- <http://byrbyr.ga/c> - *Done, and Gets Things Smart ...*
- <http://byrbyr.ga/d> - *... Because you're Smart, and Gets Things Done*
- <http://byrbyr.ga/e> - Shipping is a feature
- <http://byrbyr.ga/f> - Kernighan's Lever
- <http://byrbyr.ga/g> - Stupid technical intricacies whose “fix” isn't a “fix”
- <http://byrbyr.ga/h> - Troubleshooting software systems
- <http://byrbyr.ga/i> - Choose Boring Technology
- <http://byrbyr.ga/j> - The Unreasonable Effectiveness of Small Optimizations
- <http://byrbyr.ga/k> - Where do you test *your* code?
- <http://byrbyr.ga/l> - Antarctica
- <http://byrbyr.ga/m> - Wrap
- <http://byrbyr.ga/v> - The Art of Computer Programming (taocp)
- <https://freenom.com> - Actually free domain names

**This Slide Intentionally Left Blank**

You're Still Here?

It's over.

Go home.

Go.

