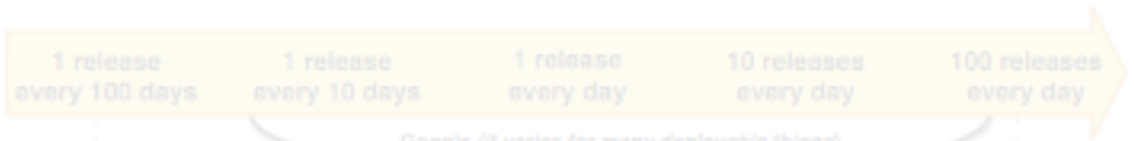


# Release frequency



# Examples



© 2014 - 2017, Paul Hammant, v2.4.2  
Let me help you migrate to Trunk-Based Development. Monorepos too, if you want to go that far.  
<https://devops.paulhammant.com>

# Branching model



This version has been superseded by another which you can find listed at: <https://devops.paulhammant.com/trunk-correlated-practices-diagram/>

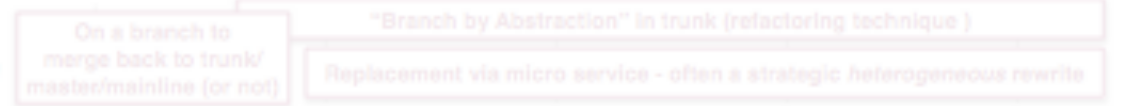
# In-house code sharing



# Use of flags/toggles



# Change that "takes a while"



# Continuous Integration



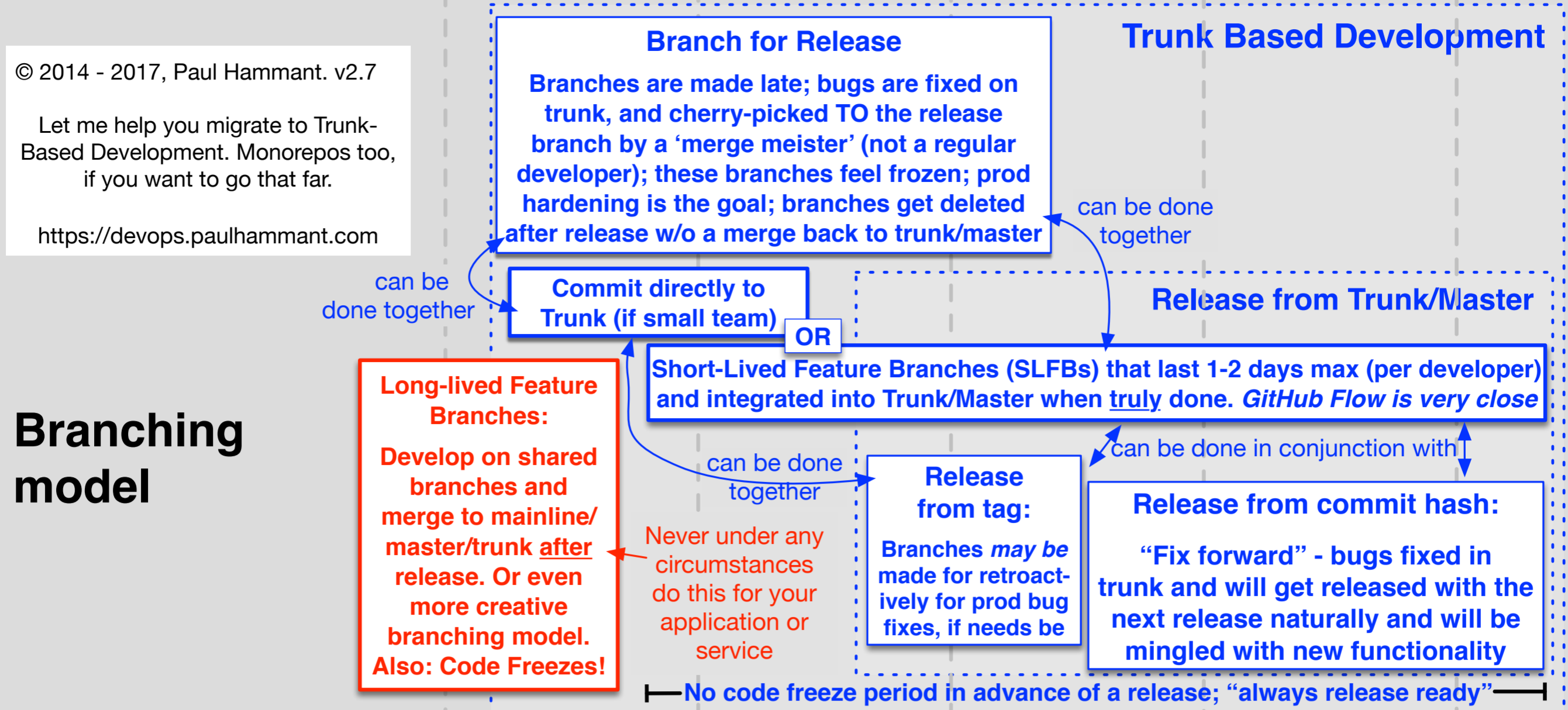
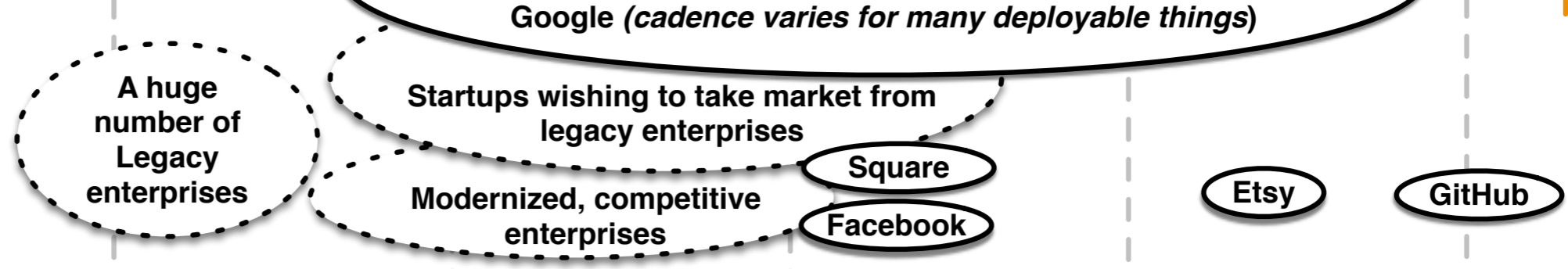
# Manual



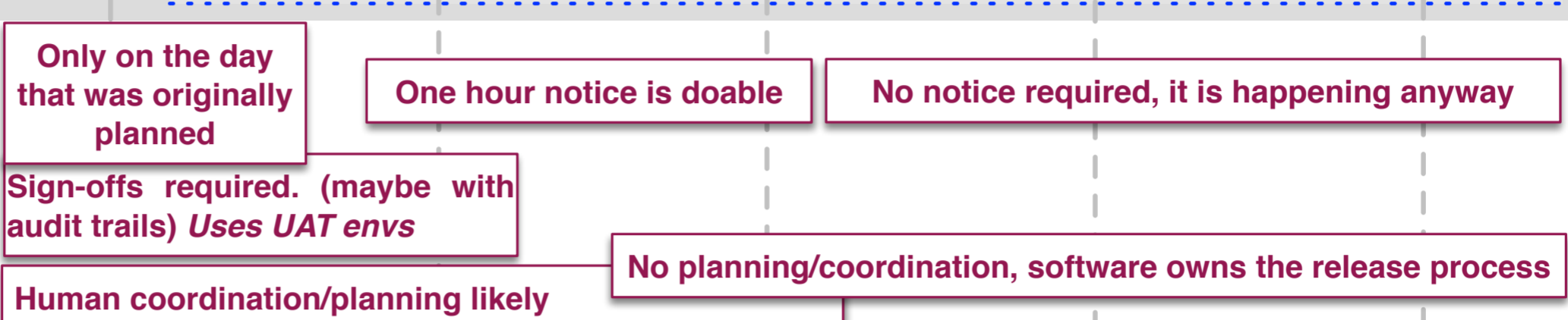
# Release frequency



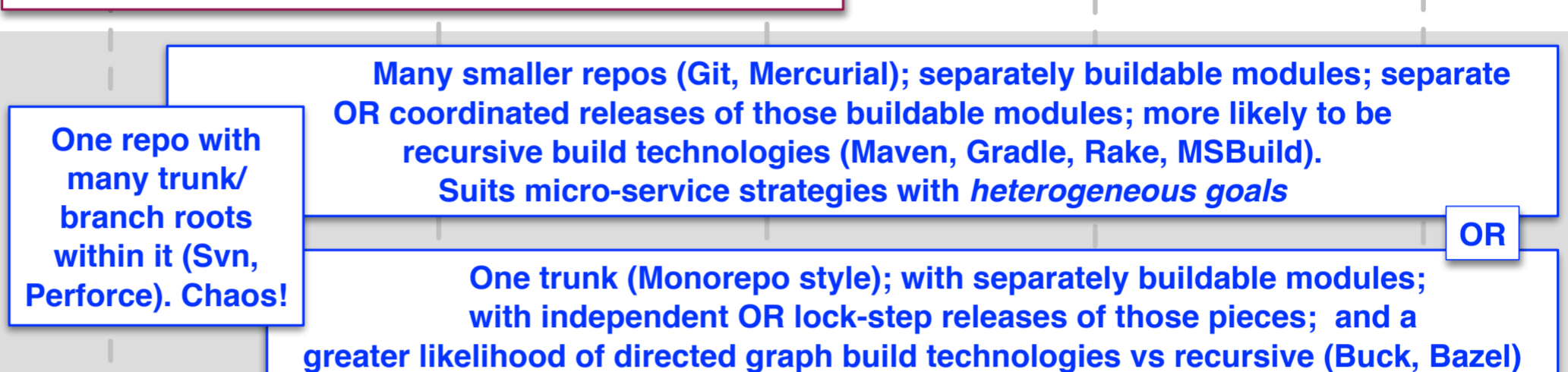
# Examples



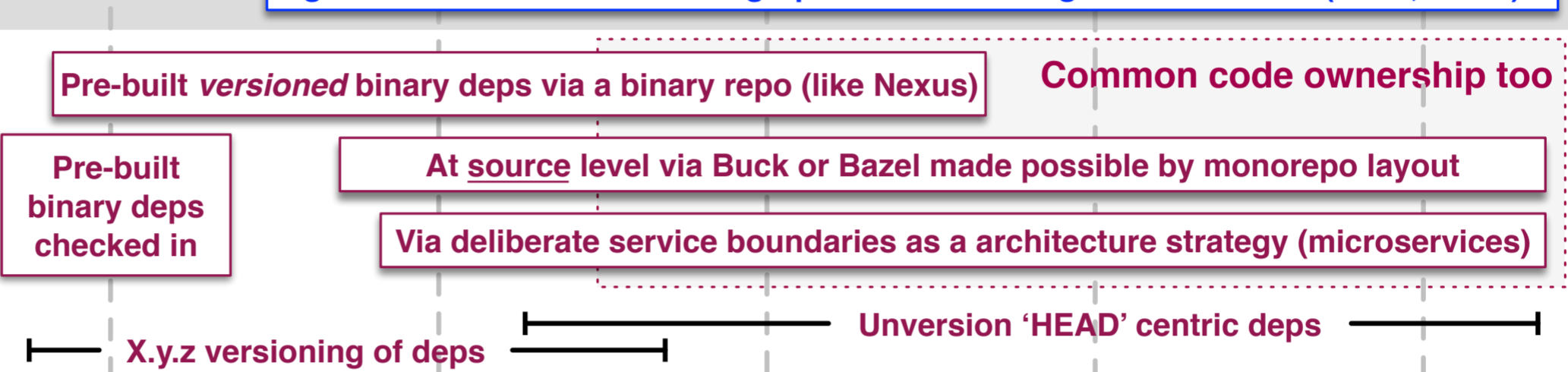
# Release preparation and readiness



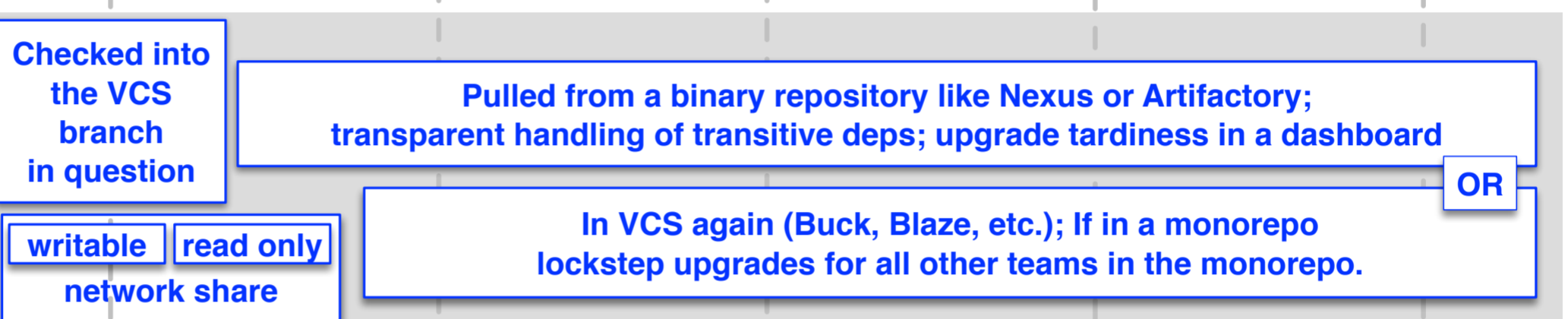
# Source Repository organization



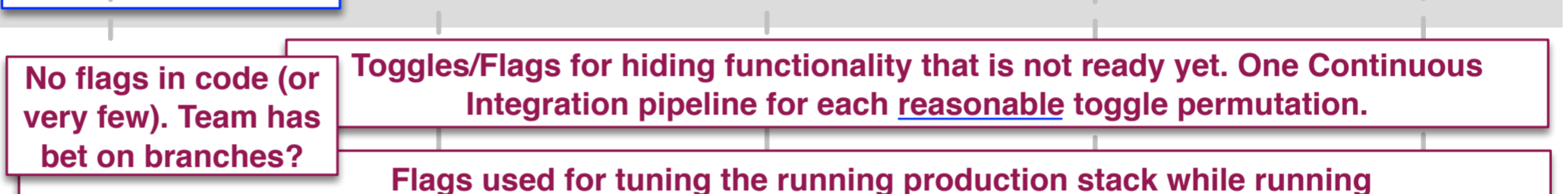
# In-house code sharing



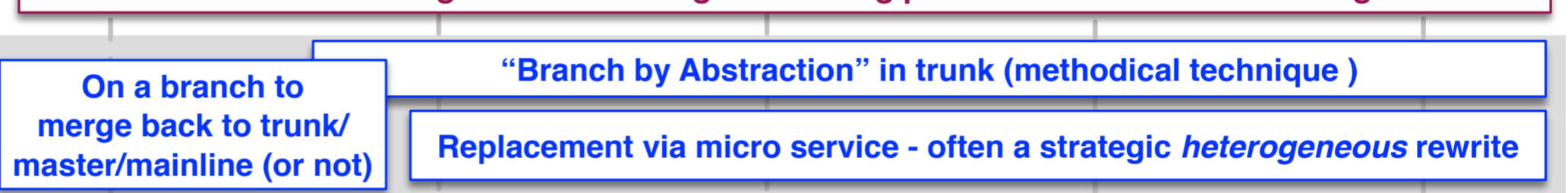
# Third-party dependencies



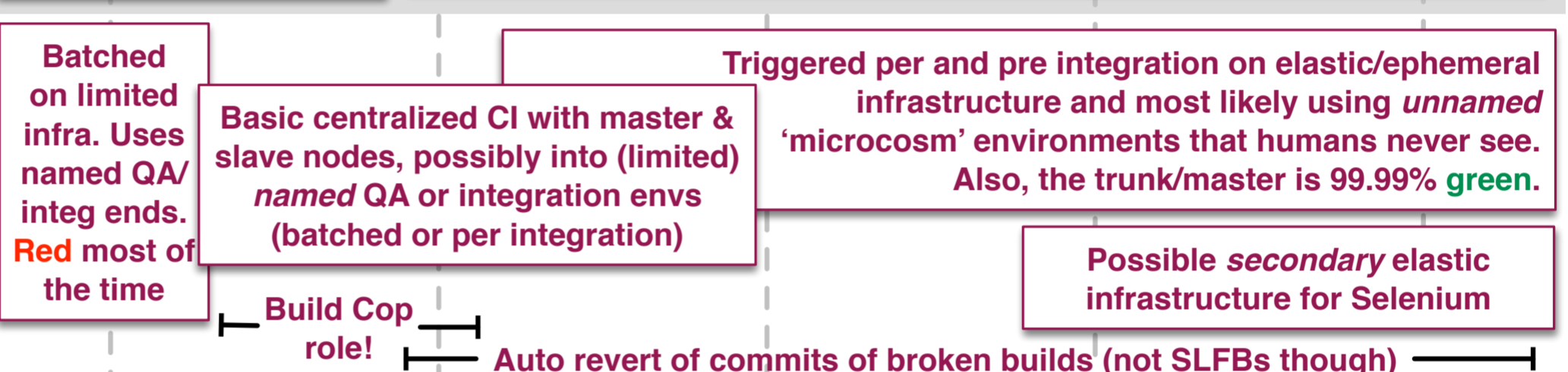
# Use of flags/toggles



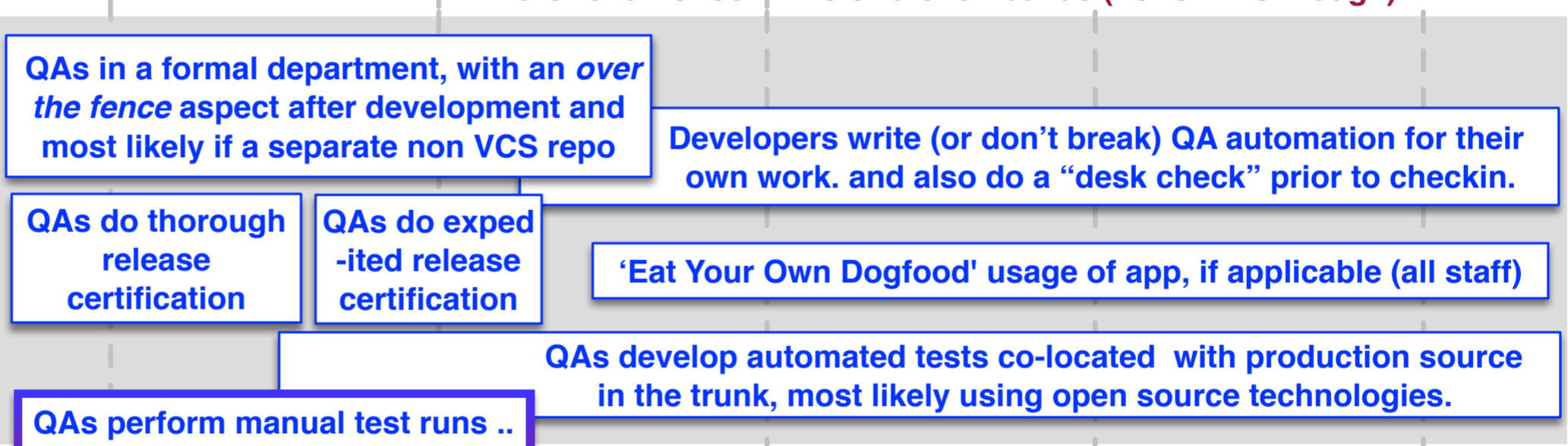
# Change that "takes a while"



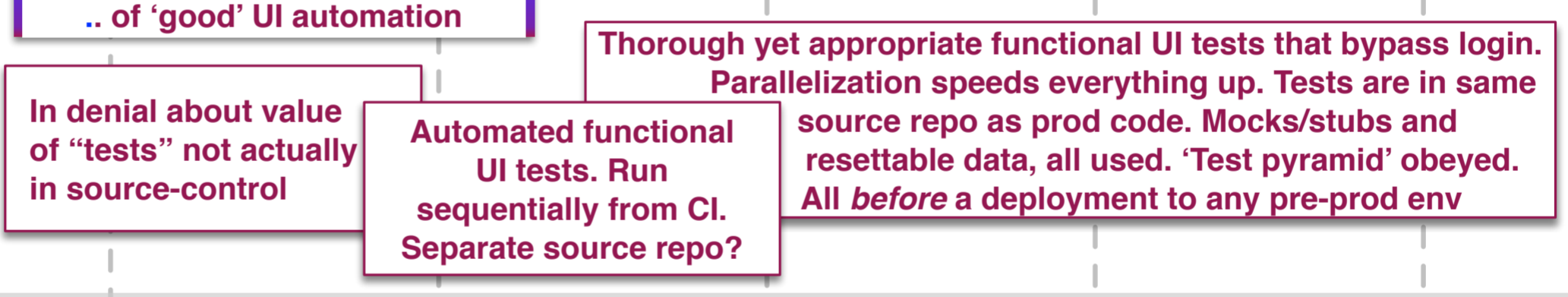
# Continuous Integration Infrastructure



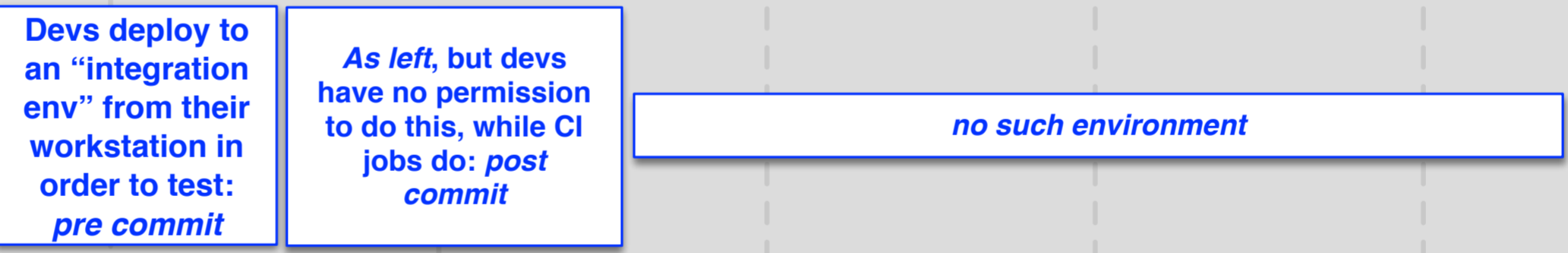
# QA activities



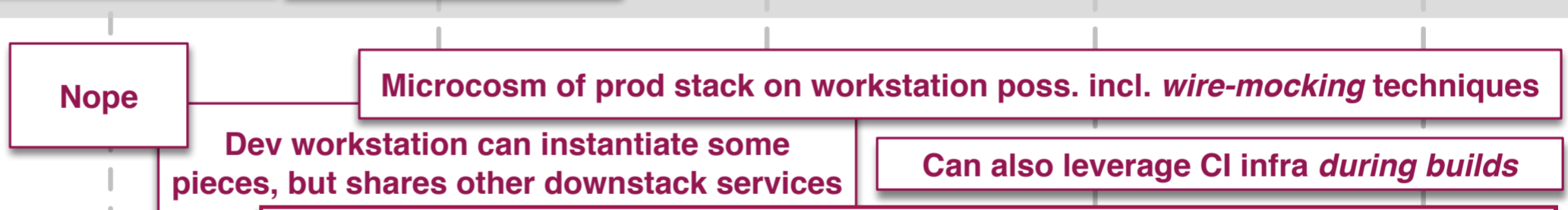
# Automated QA



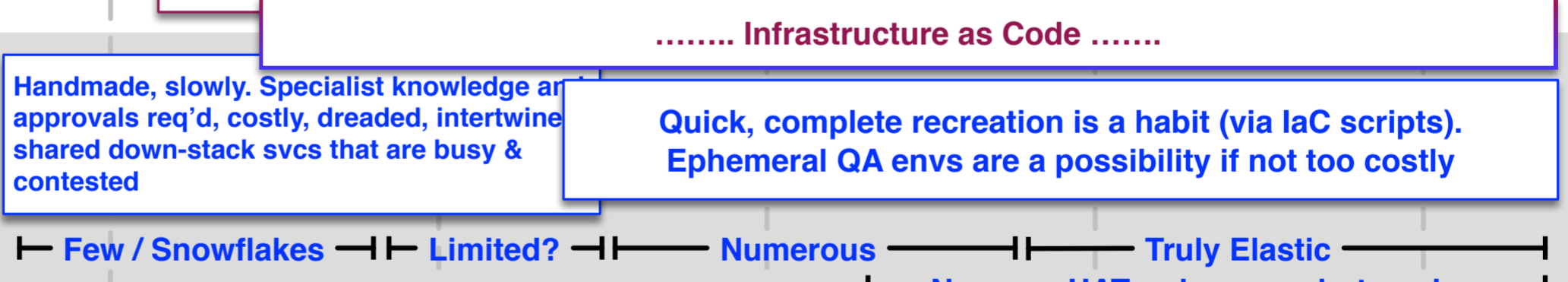
# Shared Integration testing environments (or any "for devs not QA" envs)



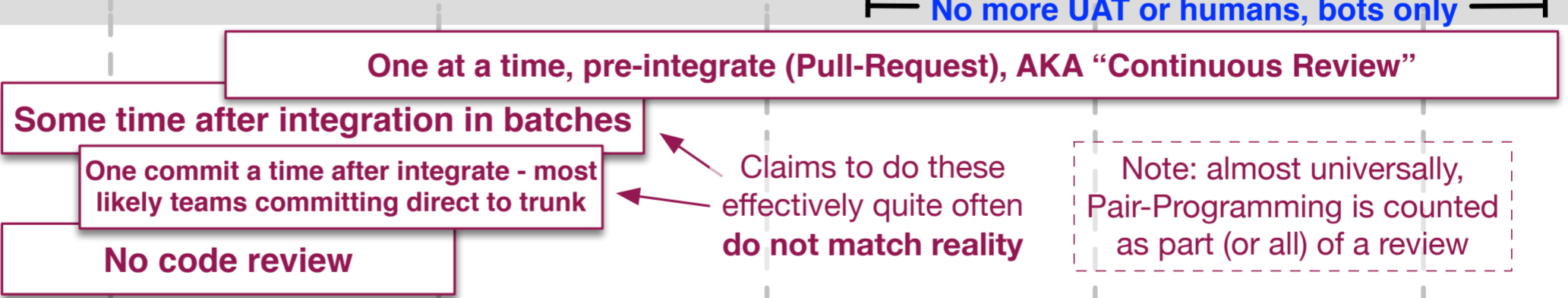
# Per-Developer Environments



# Pre-Prod environments (other than the per-dev'r ones)



# Code review



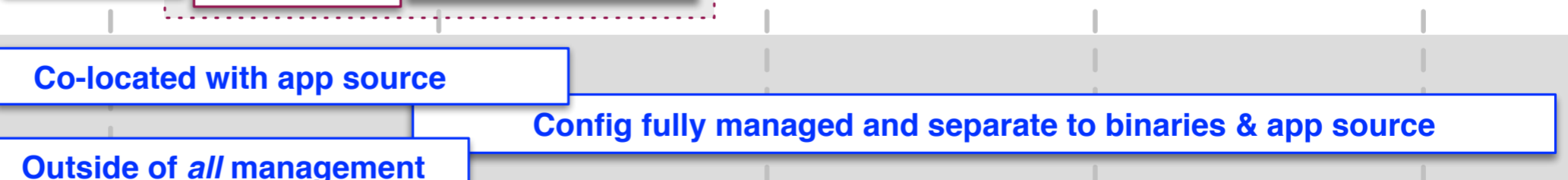
# DB rollbacks (in case of a regretted release)



# DB changes (during deployment)



# App config per Environment



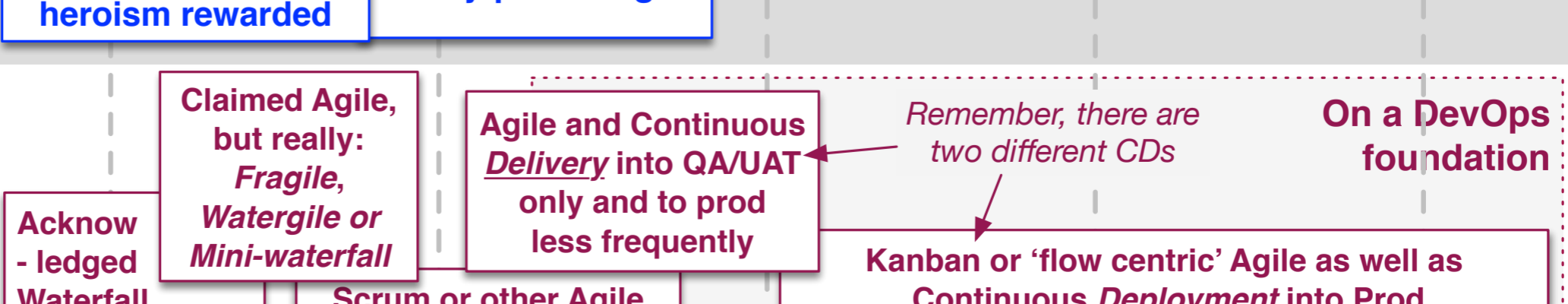
# Talent retention



# Developer activity change with proximity to release



# Methodology



# Bots make decisions for humans and execute with-out waiting/asking

