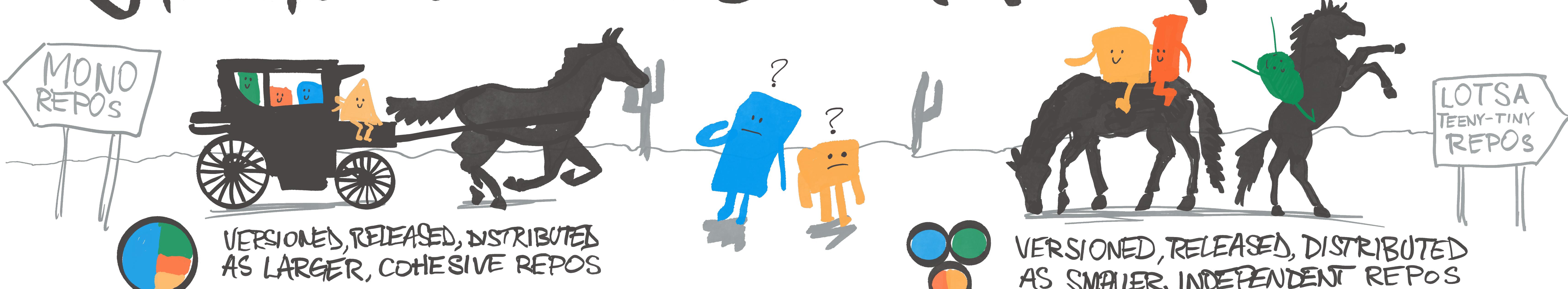
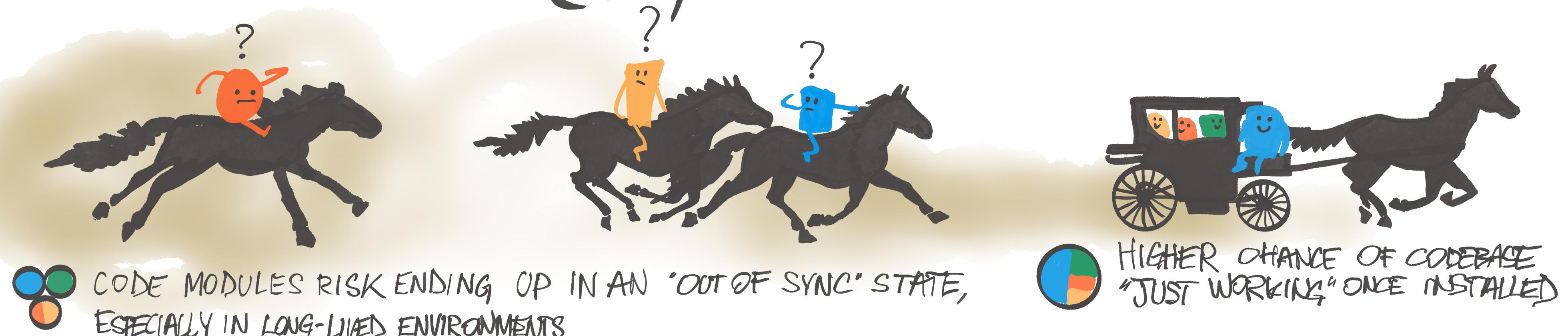


GIVEN A DIVERSE, COMPLEX CODEBASE SPANNING MULTIPLE PROJECTS, DOMAINS, AND/OR TEAMS,  
IS THERE AN **OPTIMAL REPOSITORY GRANULARITY** TO MANAGE IT  
AS IT GROWS LARGER AND MORE COMPLEX OVER TIME?

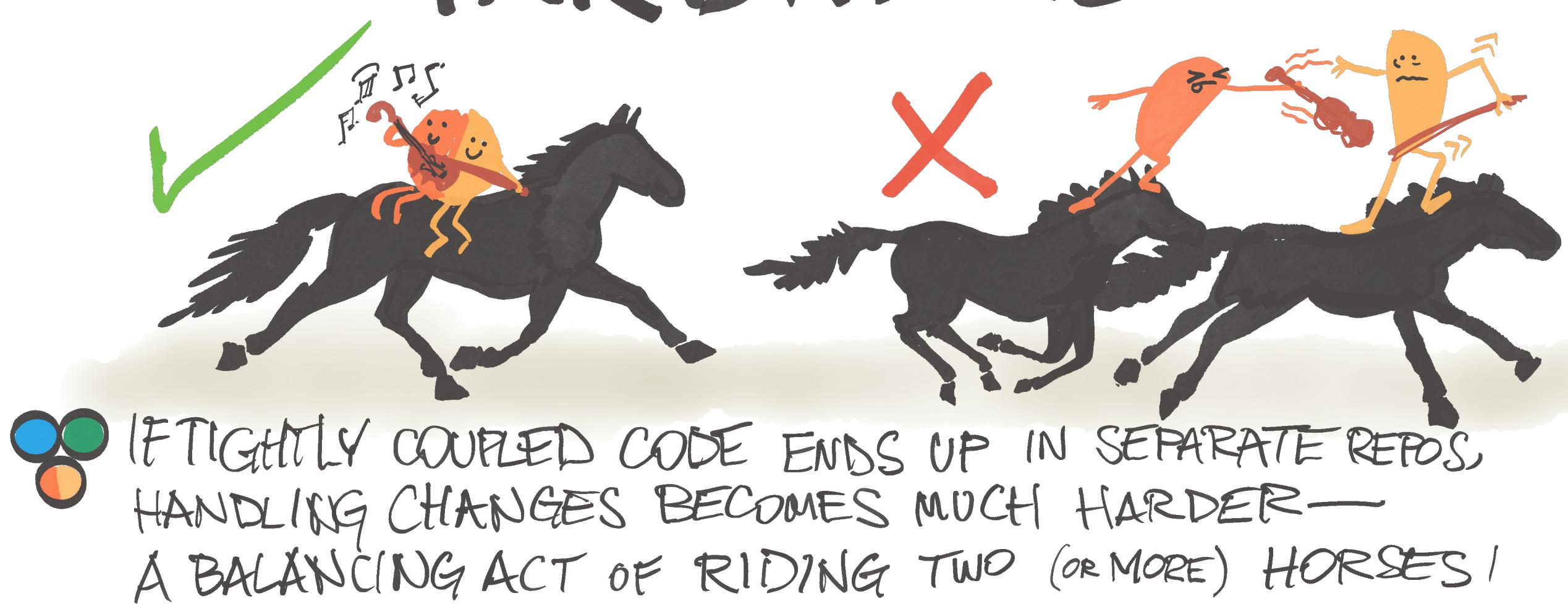
# STAGECOACHES vs HORSES



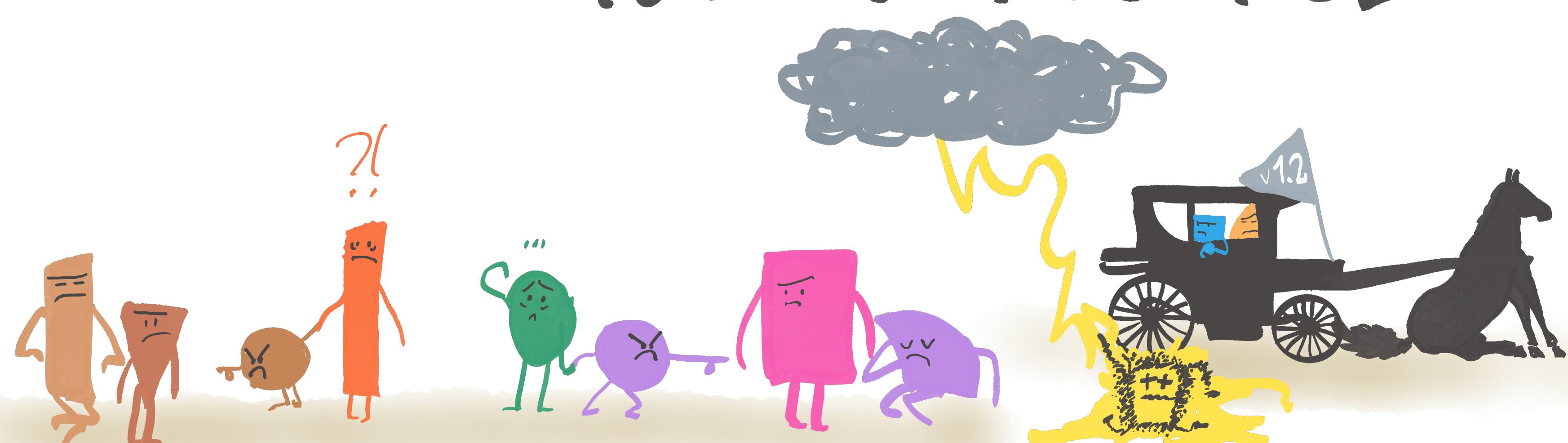
## All (?) ABOARD!



## CHOOSING THE RIGHT PARDNERS



## FOR A FISTFUL OF DEPENDENCIES



## STABLE WORK



## OUR TAKEAWAYS

- REPOS/PACKAGES ARE UNITS OF ABSTRACTION IN A CODEBASE, AND IT'S USEFUL TO TREAT THEM AS SUCH
  - MODULARITY HAS ADVANTAGES (IFF THE MODULES ARE RIGHT)
  - THE WRONG ABSTRACTION IS OFTEN MORE HARMFUL THAN NO ABSTRACTION
- EVALUATE THE TYPE, IN ADDITION TO THE AMOUNT, OF MAINTENANCE WORK
  - AD-HOC PROBLEMS AND DEPENDENCY CONFLICTS ARE LIKELY HARDER TO SOLVE